



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE141599

RANCANGAN KONTROLER PERANGKAT KERAS EH1 MILANO DENGAN MODUL WIRELESS ELECTRONICS

Idil Fitrianto
NRP 2212106035

DOSEN PEMBIMBING
Dr. Achmad Arifin, S.T., M.Eng.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE141599

**HARDWARE CONTROLLER DESIGN OF EH1 MILANO
WITH WIRELESS ELECTRONICS MODULE**

Idil Fitrianto
NRP 2212106035

Advisor
Dr. Achmad Arifin, S.T., M.Eng.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2015

**RANCANGAN KONTROLER PERANGKAT KERAS
EH1 MILANO DENGAN MODUL WIRELESS
ELECTRONICS**

TUGAS AKHIR

**Diajukan Untuk Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Elektronika
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

**Menyetujui:
Dosen Pembimbing**


Dr. Achmad Arifin, S.T., M.Eng.
NIP. 197103141997021001



Rancangan Kontroler Perangkat Keras EH1 Milano dengan Modul *Wireless Electronics*

Idil Fitrianto
2212106035

Dosen Pembimbing : Dr. Achmad Arifin ST., M.Eng

Abstrak :

Teknologi dalam dunia robot medis perkembangannya sangat pesat sekali, salah satunya adalah EH1 Milano yang dimiliki oleh salah satu laboratotium yang ada di kampus ITS Surabaya. Modul lengan robot EH1 Milano sebelumnya telah di riset oleh mahasiswa teknik elektro. Riset yang dilakukan adalah membuat kontroler robot dengan stimulus tangan manusia. Kekurangan dari riset tersebut, sistem perangkat keras masih terlalu rumit karena *interface* antara robot dan *user* masih menggunakan kabel. Selain itu keterbatasan dari sensor juga merupakan sebuah kekurangan dari riset ini. Sehingga diperlukan riset lanjutan untuk menyempurnakan riset yang telah ada. Permasalahan kerumitan perangkat keras dan keterbatasan sensor yang digunakan merupakan titik fokus pada riset ini. Sistem yang dirancang menggunakan modul *Wireless Electronics* untuk mengatasi masalah pengkabelan dan modul yang digunakan adalah modul Bluetooth model HC dengan jarak jangkauan sampai dengan 50 meter dan waktu *pairing* rata-rata 8 detik, sedangkan sensor *flex* digunakan untuk menggantikan sensor potensiometer yang digunakan sebelumnya. Sensor *flex* dengan karakteristik sensor yang linier, dengan *error percentage* terkecil sebesar 0,58% sedangkan terbesar 15,13% terhadap garis linier dapat menggantikan sensor yang sebelumnya digunakan. Untuk penelitian berikutnya dapat ditambahkan EMG sebagai pembanding impuls dari sensor *flex* dengan sinyal *electromyogram* dari tiap-tiap jari.

Kata kunci : Bluetooth HC, EH1 Milano, sensor *flex*, *wireless electronics*.

-halaman ini sengaja dikosongkan

HARDWARE CONTROLLER DESIGN OF EHI MILANO WITH WIRELESS ELECTRONICS MODULE

Idil Fitrianto
2212106035

Supervisor : Dr. Achmad Arifin ST., M.Eng

Abstract :

Medical robot technology in the world growing rapidly, one of them is EHI Milano owned by one of laboratory in Institut Teknologi Sepuluh Nopember Surabaya. Hand robot EHI Milano previously been research by electrical engineering student. Research conducted is made an robotic controller by human hand stimulus. Disadvantage of this research is, the hardware sistem is still complicated because interfacing between human and machine are connected by a wire. Further research is needed to improve the research. The complexity of hardware and the limitations of sensor become the main focus in this research. System design use Wireless Electronics module to solve the wiring problem. Wireless Electronics module in this research is Bluetooth module HC models with a range communication up to 50 meters and an average pairing time 8 seconds. Flex sensor is used to replace poetensiometer as a motion sensor. Flex sensor has a linear characteristic, with the smallest percentage error 0.58% and the highest percentage error 15.13%, against linear line. For the next research, EMG (Electromyograph) can be added as a comparison between impulses from the flex sensor and electromyogram signals from each finger.

Keyword : Bluetooth HC, EHI Milano, sensor flex, wireless electronics.

-halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT. Karena berkat rahmat-Nya penulis dapat menyelesaikan laporan tugas akhir ini. Penulis sadar bahwa dengan tanpa adanya bantuan dari pihak lain penulis tidak dapat menyelesaikan tugas akhir ini. Dengan segala hormat dan rasa rendah diri penulis mengucapkan terima kasih kepada :

1. Allah SWT. Yang telah member berkah sehat dan rahmatnya sehingga penulis dapat menyelesaikan masalah yang ditemui dalam pengerjaan tugas akhir ini.
2. Orang tua penulis, Alm. Amrizal Aziz dan mama Dasmaneli serta kakak dan adik yang selalu memberi dukungan, doa, serta semangat yang tiada hentinya kepada penulis.
3. Dosen pembimbing, Dr. Achmad Arifin., S.T., M.Eng. atas segala ilmu, bimbingan, dan pengarahannya kepada penulis dalam menyelesaikan tugas akhir ini.
4. Teman – teman konsentrasi elektronika, dan teman - teman kos ceria yang telah banyak membantu dalam pengerjaan tugas akhir ini.

Terima kasih juga pada pihak yang tidak dapat penulis sebutkan satu persatu. Penulis sadar bahwa banyaknya kekurangan dari penulis dalam penulisan laporan ini. Penulis mengharapkan masukan, kritik, dan saran untuk meningkatkan kualitas dari laporan ini dan semoga laporan ini dapat bermanfaat untuk kedepannya.

Surabaya, Januari 2015

Penulis

-halaman ini sengaja dikosongkan

DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	iii
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	1
1.3 Batasan Masalah	1
1.4 Tujuan	2
1.5 Metodologi.....	2
1.6 Sistematika Penulisan.....	2
1.7 Relevansi.....	3
BAB II.....	5
LANDASAN TEORI.....	5
2.1 Robot Tangan Prensilia EH1 Milano	5
2.1.1 Sensor pada Robot Tangan Prensilia EH1 Milano.....	5
2.1.2 Antar muka robot tangan EH1 Milano.....	7
2.2 Mikrokontroler ATMEGA8.....	8
2.2.1 Interupsi pada ATMEGA8.....	11
2.2.2 Analog to Digital Converter pada ATMEGA8	12
2.2.3 Komunikasi Serial pada ATMEGA8	14
2.3 Sensor flex.....	15
2.4 Bluetooth HC05 dan HC06	17
2.4.1 Bluetooth HC05	18
2.4.2 Bluetooth HC06	22
2.5 Voltage divider	25
2.6 Op-Amp yang digunakan sebagai buffer.....	26
BAB III	29
PERANCANGAN SISTEM	29
3.1 Perancangan Spesifikasi dari Perangkat Keras.....	29
3.1.1 Diagram blok	30
3.1.2 Data Kelistrikan Sistem	30
3.2 Perancangan Perangkat Keras	32

3.2.1	Modul Slave.....	32
3.2.2	Modul Master.....	35
3.2.3	Layout PCB	40
3.2.4	Perancangan Perangkat Lunak	41
3.3	Realisasi Alat	43
BAB IV		47
PENGUJIAN DAN ANALISA.....		47
4.1	Prosedur dan Proses Pengujian	47
4.1.1	Alat Ukur yang Digunakan	47
4.1.2	Prosedur Pengujian	49
4.1.3	Proses Pengujian	50
4.2	Hasil Pengujian	50
4.2.1	Pengujian Karakteristik Sensor Tangan Dengan Sensor potensiometermeter.....	50
4.2.2	Pengujian Karakteristik Sensor Tangan Dengan Sensor flex	53
4.2.3	Pengujian Range Jangkauan Bluetooth HC dan efektifitas komunikasi dari Bluetooth dalam melakukan komunikasi	57
4.2.4	Pengujian Rangkaian Buffer	61
4.2.5	Pengujian Test point Pada Sistem yang Dibuat	62
4.3	Analisa	64
4.3.1	Analisa Karakteristik Sensor Tangan dengan Sensor potensiometermeter.....	64
4.3.2	Analisa Karakteristik Sensor Tangan dengan Sensor flex	66
4.3.3	Analisa Range Jarak Bluetooth HC dan efektifitas komunikasi dari Bluetooth dalam melakukan komunikasi	68
4.3.4	Analisa pengujian rangkaian buffer	69
4.3.5	Analisa Test Point pada Sistem.....	69
BAB V		72
PENUTUP.....		73
DAFTAR PUSTAKA		75
LAMPIRAN.....		77
RIWAYAT PENULIS		81

DAFTAR TABEL

Tabel 2.1 Alamat motor robot EH1 Milano [1].....	7
Tabel 2.2 Data first calibration [1].....	8
Tabel 2.3 Tabel prioritas interrupt pada ATMEGA8 [2]	11
Tabel 2.4 Nilai Prescaler [2]	12
Tabel 2.5 Kombinasi pemilihan tegangan referensi ADC [2]	13
Tabel 2.6 Register mux ADC pada ATMEGA8 [2].....	13
Tabel 2.7 Contoh nilai error dengan 2 macam jenis clock [2]	15
Tabel 2.8 Keterangan pin – pin dari modul Bluetooth HC05 [4].....	20
Tabel 2.9 Keterangan PIN modul HC06	23
Tabel 2.10 Pilihan baud rate [5]	24
 Tabel 3.1 Tegangan dalam pada led.....	 33
Tabel 3.2 Pin I/O yang digunakan.....	40
Tabel 3.3 Pengiriman perintah 3 byte	42
 Tabel 4.1 Pengukuran Sensor potensiometer	 51
Tabel 4.2 Pengukuran Sensor Flex.....	54
Tabel 4.3 Pengujian jarak Bluetooth HC kondisi tanpa halangan.....	57
Tabel 4.4 Pengujian jarak Bluetooth HC kondisi dengan halangan	58
Tabel 4.5 Pengujian pengiriman data dengan halangan	60
Tabel 4.6 Pengujian pengiriman data tanpa halangan	60
Tabel 4.7 Data hasil pengujian rangkaian buffer	61
Tabel 4.8 Pengujian test point modul Master.....	63
Tabel 4.9 Pengujian test point modul Slave	63
Tabel 4.10 Data garis linier Sensor Potensiometer	64
Tabel 4.11 Data garis linier Sensor flex	66

-halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2.1 Robot tangan prensilia EH1 Milano [1]	5
Gambar 2.2 Design mekanik salah satu jari robot [1]	6
Gambar 2.3 Koneksi antara catu daya, robot EH1 Milano dan perangkat computer [1].....	7
Gambar 2.4 Pin konfigurasi ATMEGA8 [2].....	9
Gambar 2.5 Blok diagram ATMEGA8 [2]	10
Gambar 2.6 Sensor flex [3].....	16
Gambar 2.7 Simbol Potesiometer	16
Gambar 2.8 Potensiometer joystick [6].....	16
Gambar 2.9 logo Bluetooth.....	17
Gambar 2.10 Modul Bluetooth HC05 dengan adapter [4]	19
Gambar 2.11 Modul Bluetooth HC05 [4]	20
Gambar 2.12 Pin Keluaran Modul HC05 [4].....	21
Gambar 2.13 Pin Keluaran Modul USB to TTL	22
Gambar 2.14 Modul HC06 (kiri) dan rangkaian dari modul HC06 (kanan) [5].....	23
Gambar 2.15 Modul HC06S [5].....	24
Gambar 2.16 Rangkaian Pembagi Tegangan	25
Gambar 2.17 Inverting Amplifier [7].....	26
Gambar 2.18 Voltage Follower [7]	27
 Gambar 3.1 (a) Modul Bluetooth HC05, (b) Modul Bluetooth HC06, (c) Robot EH1 Milano	 29
Gambar 3.2 Diagram Blok Sistem Secara Umum.....	30
Gambar 3.3 Blok Diagram Slave	32
Gambar 3.4 Skematik Rangkaian Slave keseluruhan.....	33
Gambar 3.5 Rangkaian Indikator	34
Gambar 3.6 Blok Diagram Master	35
Gambar 3.7 Skematik Rangkaian Master keseluruhan.....	36
Gambar 3.8 Rangkaian Indikator dari Catu Batre.....	38
Gambar 3.9 Rangkaian Pembagi Tegangan Aref.....	38
Gambar 3.10 Rangkaian Pembagi Tegangan Aref.....	38
Gambar 3.11 Rangkaian ATMEGA8.....	39
Gambar 3.12 Layout sistem minimum Master ATMEGA8 V.3	41
Gambar 3.13 Layout Bluetooth Modul Slave V.3.....	41
Gambar 3.14 Diagram Blok First calibration.....	42
Gambar 3.15 Diagram Blok Tracking Sensor	42

Gambar 3.16 modul Slave yang telah di assembly	43
Gambar 3.17 Modul Master setelah diassembly	43
Gambar 3.18 Sarung Tangan Tampak dari Dalam dengan Kain Tambahan Untuk Penanam Sensor flex	44
Gambar 3.19 Sarung Tangan Tampak Dalam dengan Sensor flex yang Sudah Ditanam	44
Gambar 3.20 Sarung Tangan Tampak Dalam	45
Gambar 3.21 Sarung Tangan telah Dipasang Sensor flex	45
Gambar 3.22 Sarung Tangan Terintegrasi dengan Modul Master	46
Gambar 3.23 Modul Slave Terintegrasi dengan Robot EH1 Milano	46
Gambar 4.1 Multimeter Sanwa CD800a	47
Gambar 4.2 Osiloskop TDS2012B	48
Gambar 4.3 Meteran	48
Gambar 4.4 Stopwatch	49
Gambar 4.5 Pengujian Nilai Resistansi Sensor	50
Gambar 4.6 Karakteristik Sensor potensiometer pada Ibu Jari	51
Gambar 4.7 Karakteristik Sensor potensiometer pada Jari Telunjuk	52
Gambar 4.8 Karakteristik Sensor potensiometer pada Jari Tengah	52
Gambar 4.9 Karakteristik Sensor potensiometer pada Jari Manis	53
Gambar 4.10 Blok Pengukuran Sensor flex	53
Gambar 4.11 Karakteristik Sensor flex pada Ibu Jari	54
Gambar 4.12 Karakteristik Sensor flex pada Jari Telunjuk	55
Gambar 4.13 Karakteristik Sensor flex pada Jari Tengah	55
Gambar 4.14 Karakteristik Sensor flex pada Jari Manis	56
Gambar 4.15 Karakteristik Sensor flex pada Jari Kelingking	56
Gambar 4.16 Blok Pengukuran Jangkauan Bluetooth HC	57
Gambar 4.17. Monitoring penerimaan data dari modul wireless dengan menggunakan RealTime Serial Capture Program.	59
Gambar 4.18. Grafik pengujian rangkaian buffer	62
Gambar 4.19 Pengujian test point pada modul Master	62
Gambar 4.20 Pengujian test point pada modul Slave	63

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan dunia robot medis sangat pesat sekali, hampir setiap peran manusia pada dunia industri telah digantikan oleh peran robot guna mengurangi tingkat kecelakaan dalam dunia kerja. Robot EH1 Milano merupakan potongan organ tangan yang menyerupai tangan manusia, yang dilengkapi dengan lima jari dan fungsinya sama dengan jari manusia pada umumnya. Kekuatan cengkraman dari jari robot pun juga dibuat agar mendekati bentuk organ tangan sebenarnya, dengan menggunakan sensor arus yang di tanamkan pada robot.

Robot tangan ini dapat dijumpai pada salah satu laboratorium teknik elektro ITS, yang sebelumnya telah di riset untuk mengontrol robot tersebut. Pengontrolan dibuat dengan *input* tangan manusia yang telah dilengkapi dengan sensor guna memastikan perintah yang akan dikirim ke *Sensor Proximity* EH1 Milano. Pada robot EH1 Milano dilengkapi dengan sensor – sensor pendukung yang digunakan untuk menggerakkan seluruh segmen dari tiap jarinya. Hasil dari riset ini sangat baik, namun masih ada beberapa kekurangan yaitu karena komunikasi dari user ke robot masih menggunakan kabel sehingga bisa dibilang kurang praktis.

Dari hasil riset sebelumnya itu, maka akan dilakukan riset lanjutan guna mengembangkan sistem yang telah dibuat sebelumnya menjadi sistem tanpa kabel dengan tujuan portabilitas.

1.2 Perumusan Masalah

Permasalahan yang dibahas dalam tugas akhir ini adalah:

1. Bagaimana merancang perangkat keras EH1 Milano?
2. Apakah sistem *wireless* dapat diterapkan dalam sistem kontrol robot EH1 Milano?
3. Bagaimana karakteristik dari *Sensor flex* yang digunakan sebagai *input* pada robot EH1 Milano?

1.3 Batasan Masalah

Batasan masalah dalam tugas akhir ini adalah:

1. Perancangan sistem perangkat keras EH1 Milano.
2. Membandingkan kinerja *Sensor flex* dengan Sensor potensiometermeter.

3. Menguji jarak jangkauan dari modul *wireless* yang digunakan.

1.4 Tujuan

Tujuan penelitian pada tugas akhir ini adalah sebagai berikut:

1. Membuat sistem kendali tanpa kabel.
2. Sistem kendali yang dibuat sangat sederhana dengan tujuan portabelitas.

1.5 Metodologi

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi Literatur
Pada tahap ini dilakukan pengumpulan dasar teori yang mendukung dalam penulisan tugas akhir. Dasar teori dapat diambil dari buku-buku, jurnal, dan artikel – artikel dari internet.
2. Perancangan Perangkat Keras
Pada tahap ini dirancang perangkat keras yang terdiri dari *board* mikrokontroler ATMEGA8, modul *Bluetooth* untuk komunikasi tanpa kabel, dan sarung tangan yang sudah ditanamkan *Sensor flex*.
3. Perancangan Perangkat Lunak
Perancangan perangkat lunak meliputi pembuatan *firmware* untuk mikrokontroler yang digunakan untuk menguji sistem yang telah dirancang.
4. Pengujian Sistem
Proses pengujian sistem dilakukan dengan membandingkan karakteristik dari *Sensor flex* sebagai sensor *input*, dan sensor yang digunakan sebelumnya yaitu potensiometer.
5. Penulisan Laporan Tugas Akhir
Tahap penulisan laporan adalah tahap akhir dari proses pengerjaan tugas akhir. Isi laporan berisikan hal yang bersangkutan dengan tugas akhir yang dikerjakan. Meliputi pendahuluan, teori penunjang, perancangan sistem, pengujian, dan penutup.

1.6 Sistematika Penulisan

Untuk pembahasan lebih lanjut, laporan tugas akhir ini disusun dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi pendahuluan yang membahas latar belakang, perumusan masalah, batasan masalah, tujuan, metodologi, sistematika pembahasan, dan relevansi.

BAB II TEORI PENUNJANG

Bab ini berisi teori penunjang dan literatur yang berguna bagi pembuatan tugas akhir ini. Teori penunjang yang diberikan antara lain teori mengenai *Sensor flex*, *data sheet* modul *Bluetooth*, dan *data sheet* ATMEGA8.

BAB III PERANCANGAN SISTEM

Bab ini berisi perencanaan sistem untuk realisasi sistem kendali robot secara *wireless*. Perancangan sistem terdiri dari perancangan perangkat keras dan perencanaan perangkat lunak.

BAB IV PENGUJIAN DAN ANALISA

Bab ini berisi hasil pengujian dalam tugas akhir ini, dan disertai beberapa analisa yang berkaitan dengan pengujian yang dilakukan.

BAB V PENUTUP

Penutup yang berisi kesimpulan yang diperoleh dari pembuatan tugas akhir ini, serta saran-saran untuk pengembangan lebih lanjut.

1.7 Relevansi

Hasil yang diperoleh dari tugas akhir ini diharapkan dapat bermanfaat untuk pengembangan kedepan dalam hal pengaplikasian robot tangan untuk menggantikan peran tangan manusia dalam melakukan aktifitas yang memiliki resiko kecelakaan sangat tinggi.

-halaman ini sengaja dikosongkan

BAB II

LANDASAN TEORI

2.1 Robot Tangan Prensilia EH1 Milano

Robot tangan prensilia EH1 Milano merupakan sebuah perangkat tangan buatan yang menyerupai bentuk dari tangan manusia. Robot tangan ini dapat diaplikasikan pada bidang biomedik yang menggantikan bagian organ tubuh khususnya bagian tangan. Robot ini terdiri dari beberapa bagian diantaranya : sensor, kendali internal, dan antar muka dengan perangkat komputer atau kendali lain dari luar robot.

2.1.1 Sensor pada Robot Tangan Prensilia EH1 Milano

Berikut merupakan sensor – sensor yang terpasang pada perangkat robot Prensilia EH1 Milano :

a) Rotary encoder

Sensor *Rotary encoder* ini digunakan untuk mengontrol posisi dari jari oleh putaran motor pada robot. Pada robot prensilia EH1 Milano terdapat enam buah sensor yang terpasang pada bagian robot. Diantaranya adalah : dua buah sensor pada bagian ibu jari, satu buah pada bagian telunjuk, satu buah pada bagian jari tengah, satu buah untuk bagian jari manis dan satu buah untuk jari kelingking.



Gambar 2.1 Robot tangan prensilia EH1 Milano [1]

b) Sensor Arus

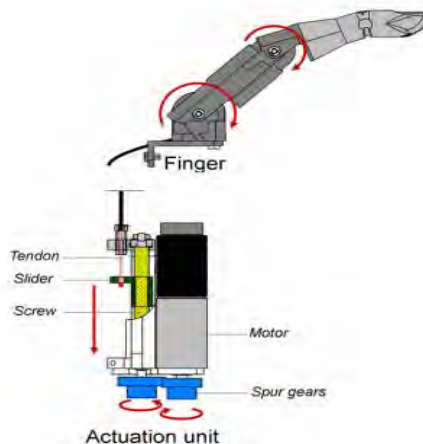
Sensor arus pada bagian robot prensilia EH1 Milano digunakan untuk mengukur kekuatan genggamannya dari robot pada saat mencengkram sesuatu. Dengan kata lain pada waktu motor penggerak robot berputar namun ditahan oleh sesuatu maka motor akan membutuhkan arus yang besar. Terdapat enam buah sensor arus yang terpasang pada tiap jari robot diantaranya : dua buah sensor pada bagian ibu jari, satu buah pada bagian telunjuk, satu buah pada bagian jari tengah, satu buah untuk bagian jari manis dan satu buah untuk jari kelingking.

c) Sensor Ketegangan

Terdapat lima buah sensor ketegangan pada robot prensilia EH1 Milano. Diantaranya : dua buah pada bagian ibu jari, satu buah pada bagian telunjuk, satu bagian pada jari tengah, dan satu buah pada bagian jari manis dan satu buah untuk jari kelingking.

d) *Sensor Proximity*

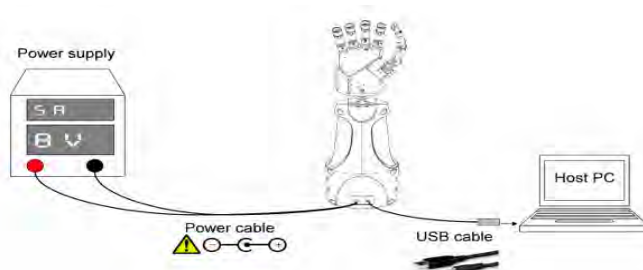
Ada 12 buah *Sensor Proximity* yang terpasang robot yang berfungsi untuk mengetahui posisi dari jari, apakah terbuka penuh atau tertutup. Sensor ini dipasang dua buah untuk tiap jari. Empat buah pada bagian ibu jari, dua buah untuk jari telunjuk, dua buah untuk jari tengah, dan dua buah untuk jari manis dan dua buah untuk jari kelingking.



Gambar 2.2 Design mekanik salah satu jari robot [1]

2.1.2 Antar muka robot tangan EH1 Milano

Perangkat perangkat yang terpasang pada bagian robot tangan prensilia dapat diakses dengan komunikasi serial UART (*Universal Asynchronous Receiver Transmitter*). Dengan *baud rate* 1152000 bps, data bits 8 dan *stop* bit 1. Sumber tagangan untuk robot prensilia yaitu +8V dengan arus maksimum 3A pada saat semua motor bergerak dengan beban maksimum. Untuk melakukan antar muka dengan robot EH1 Milano dibutuhkan satu atau beberapa *byte* data yang dikirimkan dari mikrokontroler maupun komputer dengan format yang berbeda tergantung perintah yang digunakan. Perintah yang dikirim terbagi menjadi dua, yaitu : LLMC (*Low Level Motion Control*) dan HLHC (*High Level Hand Controler*). Perbedaan dari keduanya adalah perintah pengiriman datanya, LLMC minimal mengirim sebanyak 5 *byte* data sedangkan HLHC minimal mengirim 1 *byte* data. Dari kedua kondisi diatas jelas terlihat HLHC dapat melakukan proses pengiriman yang lebih cepat dibandingkan dengan LLMC. Berikut ditunjukkan pada **Gambar 2.3**.



Gambar 2.3 Koneksi antara catu daya, robot EH1 Milano dan perangkat komputer [1]

Tabel 2.1 Alamat motor robot EH1 Milano [1]

Motor Address	Jari
0	Ibu jari <i>abduction/adduction</i>
1	Ibu jari
2	Telunjuk
3	Jari tengah
4	Jari manis
5	Jari kelingking

Tabel 2.2 Data first calibration [1]

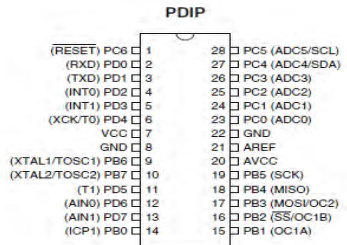
<i>Byte-1</i>						
0	1	0	0	0	0	1

Format pengiriman data yang digunakan adalah HLHC dengan minimal mengirim 1 *byte* data. Contoh pengiriman data HLHC 1-*byte* ini adalah kalibrasi awal (*first calibration*). *First calibration* (FiC), kalibrasi awal ini mengirimkan data 1-*byte* yaitu 66 *decimal* (01000010_b) dan tidak ada data *feed back* dari robot.

2.2 Mikrokontroler ATMEGA8

Mikrokontroler merupakan perangkat kontroler yang telah dilengkapi dengan fitur-fitur yang cukup lengkap seperti Unit pengolahan data, fitur ADC, *timer*, memori *flash*, *eeprom*, dan lain sebagainya. Untuk tugas akhir ini penulis menggunakan mikrokontroler jenis ATMEGA8 produk dari atmel. Mikrokontroler ATMEGA8 ini dapat digunakan sebagai media pembantu dalam bidang kontroler karena fitur-fiturnya yang sudah cukup banyak. Adapun fitur-fitur yang telah disediakan oleh mikrokontroler ATMEGA8 ini adalah sebagai berikut:

- ATMEGA8 memiliki jalur data dengan lebar sebesar 8-bit.
- Jumlah pin I/O pada ATMEGA8 ada 23 dari total 28 pin yang 5 lainnya merupakan pin untuk catu daya dan tegangan referensi.
- Memory *flash* ROM sebesar 8 KB, 1Kb SRAM, dan 512 *byte EEPROM* yang dapat dimanfaatkan sesuai dengan kebutuhan.
- Komunikasi serial dengan mikroprosesor atau perangkat lain dengan menggunakan UART yang telah tersedia pada ATMEGA8.
- 2 pin *interrupt* eksternal yang dapat digunakan dalam aplikasi tertentu dengan mengubah mode pin tersebut menjadi fitur *interrupt* namun ada juga *interrupt* internal yang tersedia pada mikroprosesor.
- Memiliki 6-pin ADC yang dapat diatur resolusinya 8-bit atau 10-bit.
- 3-pin *timer/counter* yang dapat difungsikan menjadi PWM (*pulse width modulation*) dengan resolusi 8-bit atau 10-bit.
- Tegangan kerja 2,7V – 5,5V untuk seri ATMEGA8L dan 4,5V – 5,5V untuk seri ATMEGA8.

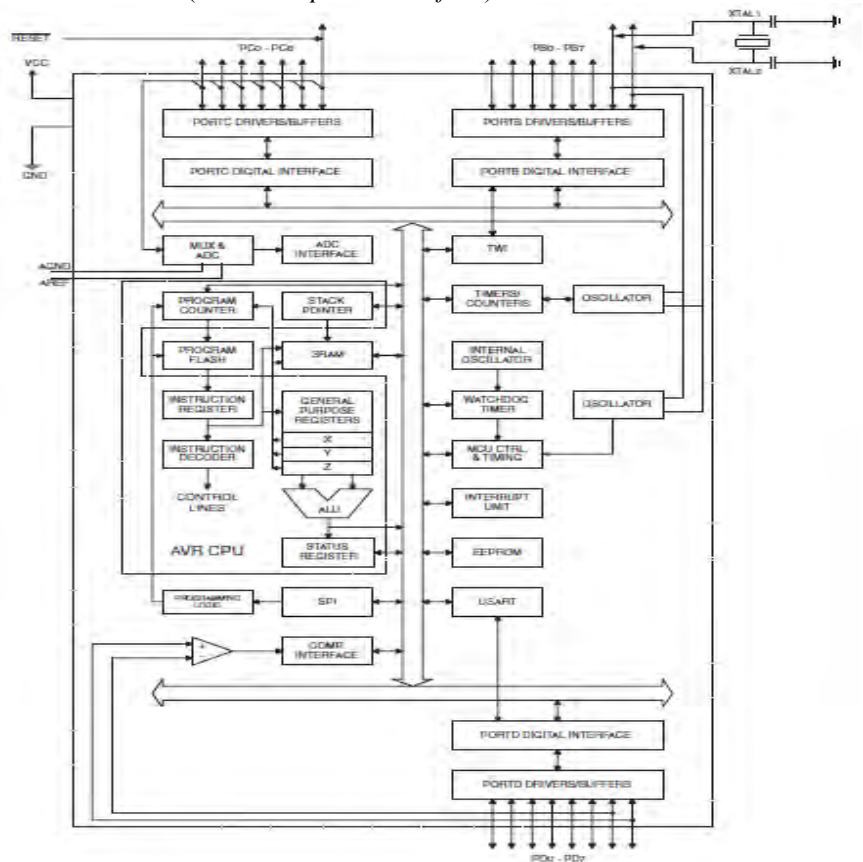


Gambar 2.4 Pin konfigurasi ATMEGA8 [2]

Gambar 2.4 merupakan pin konfigurasi dari mikroprosesor yang terdapat 28 pin yang memiliki fitur masing-masing dari tiap pin. Berikut ini merupakan keterangan dari konfigurasi pin ATMEGA8 diatas :

- VCC, merupakan port catu daya untuk mengoperasikan mikrokontroler.
- GND, merupakan referensi tegangan pada mikrokontroler.
- AVCC, merupakan catu daya analog untuk ADC pada mikroprosesor.
- AREF, tegangan referensi untuk ADC.
- PORTB(PB0 - PB7), merupakan port I/O digital yang telah dilengkapi dengan resistor *pull up* internal.
- PB0(ICP1), merupakan *input trigger* untuk *timer 1*.
- PB1-PB3(OC1A,OC1B,OC2), merupakan keluaran PWM pada *timer 1* dengan resolusi 8-bit atau 10-bit.
- PB6-PB7, memiliki fungsi khusus yaitu untuk mode *external clock* dikoneksikan pada port ini, dengan maksimal sebesar 16MHz.
- PORTC(PC0 - PC5), merupakan port ADC dengan resistor *pull up* internal, pada port ini dapat menerima masukan berupa sinyal analog.
- PC6/Reset, merupakan port untuk me-Reset sistem pada mikroprosesor secara keseluruhan.
- PORTD(PD0 - PD7), merupakan port I/O digital yang dilengkapi dengan resistor *pull up* internal.
- PD0(Rx), memiliki fungsi khusus untuk menerima data pada komunikasi serial.
- PD1(Tx), memiliki fungsi khusus untuk mengirim data pada komunikasi serial.

- n) PD2-PD3(INT0 - INT1), memiliki fungsi khusus yaitu untuk menginterrupt program secara eksternal atau secara *hardware* dari luar.
- o) PD4(XCK), merupakan port yang berfungsi sebagai *clock* I/O pada USART eksternal dan juga sebagai *timer/counter* 0 eksternal.
- p) PD5, merupakan port *timer/counter* 1 eksternal.
- q) PD6-PD7(AIN0 – AIN1), merupakan masukan pembanding analog positif(AIN0), masukan pembanding analog negatif(AIN1).
- r) SCK,MOSI,MISO,SS, merupakan pin untuk melakukan komunikasi serial SPI (*Serial Peripheral Interface*).



Gambar 2.5 Blok diagram ATMEGA8 [2]

Pada **Gambar 2.5** menunjukkan tentang blok diagram dari ATMEGA8, dari gambar tersebut terlihat bahwa fitur dari ATMEGA8 yang cukup kompleks seperti yang telah dijelaskan sebelumnya.

2.2.1 Interupsi pada ATMEGA8

Interupsi merupakan suatu cara untuk menghentikan alur dari program utama dan dialihkan ke program *interrupt*. Program yang sedang berjalan pada saat terkena interupsi akan terhenti, pindah ke program interupsi, dan apabila interupsi telah selesai dieksekusi maka akan kembali pada program utama yang terhenti tadi dan melanjutkan siklus seperti biasa kembali. Interupsi pada ATMEGA8 terdapat vektor prioritas, dengan yang tertinggi adalah 0 yaitu *Reset*. Prioritas interupsi maksudnya siklus interupsi yang akan dieksekusi adalah dimulai dari prioritas tertinggi dan setelah selesai akan berpindah ke interupsi dengan prioritas lebih rendah. Interupsi dilakukan hanya bila perintah interupsi ini dipanggil, dengan tujuan agar tidak melakukan pengecekan yang berulang secara terus menerus melainkan hanya berdasarkan permintaan interupsi saja. Berikut merupakan tabel interupsi dengan prioritas tertinggi sampai dengan terendah.

Tabel 2.3 Tabel prioritas interrupt pada ATMEGA8 [2]

Prioritas vektor	Alamat vektor	Sumber	Keterangan
1	0x000	<i>RESET</i>	<i>RESET</i>
2	0x001	INT0	Interupsi 0
3	0x002	INT1	Interupsi 1
4	0x003	<i>TIMER2 COMP</i>	<i>Timer 2 Compare cocok</i>
5	0x004	<i>TIMER2 OVF</i>	<i>Timer 2 Over Flow</i>
6	0x005	<i>TIMER1 CAPT</i>	<i>Timer 1 Capture Event</i>
7	0x006	<i>TIMER1 COMPA</i>	<i>Timer 1 CompareA cocok</i>
8	0x007	<i>TIMER1 COMPB</i>	<i>Timer 1 CompareB cocok</i>
9	0x008	<i>TIMER1 OVF</i>	<i>Timer 1 Over Flow</i>
10	0x009	<i>TIMER0 OVF</i>	<i>Timer 0 Over Flow</i>
11	0x00A	SPI, STC	Transfer Serial Selesai
12	0x00B	USART, RXC	USART, Rx Selesai
13	0x00C	USART, UDRE	USART, Data Register Kosong
14	0x00D	USART, TXC	USART, Tx Selesai
15	0x00E	ADC	Konversi ADC Selesai
16	0x00F	EE_READY	<i>EEPROM</i> Siap
17	0x010	ANA_COMP	Analog Comparator

18	0x011	TWI	<i>Two Wire Interface</i>
19	0x012	SPM_RDY	Store Program Memori Siap

Contoh penggunaan interupsi ini yaitu, pada waktu sistem sedang berjalan yang apabila kita tekan *RESET* atau memberi logika *low* pada pin *RESET* maka program akan terhenti dan mengulang kembali dari awal.

2.2.2 Analog to Digital Converter pada ATMEGA8

Analog to digital converter diperlukan untuk mengolah sinyal *input* yang berupa analog. ADC pada mikrokontroler ini memiliki resolusi 8-bit untuk pembacaan 0 – 255 atau 10-bit untuk pembacaan 0 - 1023. Jenis ADC yang digunakan pada mikroprosesor ATMEGA8 adalah *successive approximation* atau dengan kata lain ADC ini menggunakan teknik pendekatan dengan *clock* yang digunakan sebesar 50kHz – 200kHz. Untuk memaksimalkan penggunaan ADC ini dianjurkan untuk menggunakan nilai di antara *range* diatas untuk hasil yang maksimal. Sumber *clock* yang digunakan pun merupakan sumber dari pendekatan (*approximation*). Pendekatan ini dilakukan sebanyak 13 siklus sebelum akhirnya menjadi nilai ADC yang sebenarnya. Dengan demikian frekuensi sampling dari ADC dapat dihitung dengan Persamaan :

$$f_{smp} = \frac{f_{clock}}{13N} \quad (2.1)$$

Keterangan :

f_{smp} : Frekuensi sampling.

f_{clock} : *Clock* yang digunakan oleh mikrokontroler.

N : *Prescaler* (pembagi frekuensi).

Dari Persamaan diatas dapat kita ketahui nilai dari frekuensi samplingnya ADC. Untuk nilai *Prescaler* dapat dilihat pada **Tabel 2.4** berikut ini :

Tabel 2.4 Nilai *Prescaler* [2]

ADPS2	ADPS1	ADPS0	Nilai <i>Prescaler</i> (N)
0	0	0	2
0	0	1	2

0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Tabel 2.5 Kombinasi pemilihan tegangan referensi ADC [2]

REFS1	REFS0	Jenis Tegangan Referensi
0	0	AREF, Vref Internal dinon-aktifkan
0	1	AVCC dengan kapasitor eksternal pada pin AREF
1	0	-
1	1	Internal dengan kapasitor eksternal pada pin AREF

Untuk pemilihan tegangan referensi ADC dilakukan dengan mengatur nilai register ADMUX pada kombinasi bit REFS1 dan REFS0. Pengaturan register ini ditunjukkan pada **Tabel 2.5**. Tegangan referensi yang dapat digunakan untuk ADC yaitu ada 3 diantaranya :

- AVCC, *Inputan* tegangan eksternal dengan nilai 0 – 5V
- AREF, *Inputan* tegangan eksternal dengan nilai 0 – 5V
- Tegangan referensi internal yang sudah disediakan oleh mikrokontroler dengan nilai 2,56V

ATMEGA8 memiliki 6 kanal pin ADC yang dipilih menggunakan multiplexer 4-bit untuk setiap kanal yang hendak dibaca. Resolusi pembacaan ADC ini yaitu 8-bit atau 10-bit. Oleh karena *inputan* ADC ini diatur dengan menggunakan multiplexer, sehingga konfigurasi pengaturannya dapat dilihat pada **Tabel 2.6** berikut ini.

Tabel 2.6 Register mux ADC pada ATMEGA8 [2]

Bit ke-3	Bit ke-2	Bit ke-1	Bit ke-0	Masukan ADC
0	0	0	0	ADC0
0	0	0	1	ADC1
0	0	1	0	ADC2
0	0	1	1	ADC3
0	1	0	0	ADC4
0	1	0	1	ADC5
0	1	1	0	ADC6

0	1	1	1	ADC7
1	0	0	0	-
1	0	0	1	-
1	0	1	0	-
1	0	1	1	-
1	1	0	0	-
1	1	0	1	-
1	1	1	0	1,3V
1	1	1	1	GND

Sistem kerja dari ADC ini yaitu tegangan yang masuk ke multiplekser akan di umpankan ke komparator dan dibandingkan dengan nilai DAC internal dengan resolusi 8-bit atau 10-bit tergantung yang mana yang digunakan. Kemudian nilainya dibandingkan lagi ke ADC agar nilai yang dikeluarkan oleh ADC merupakan nilai pendekatan sesuai dengan *inputan* pada multiplekser sebelumnya. Untuk menghitung nilai hasil konversi ADC dapat menggunakan Persamaan berikut ini :

$$ADC_{nilai} = \frac{V_{in} \cdot 2^{resolusi}}{V_{ref}} \quad (2.2)$$

Keterangan :

ADC_{nilai} : Nilai ADC.

V_{in} : Tegangan *input* pada pin ADC.

Resolusi : 8-bit atau 10-bit.

V_{ref} : tegangan referensi ADC.

2.2.3 Komunikasi Serial pada ATMEGA8

Pada ATMEGA8 dapat melakukan komunikasi serial baik mengirim maupun menerima data dengan menggunakan perangkat UART (*Universal Asynchronous Receiver Transmitter*). Pin yang digunakan untuk komunikasi ini yaitu pin Tx (PD1) sebagai pengirim data dan Rx (PD0) sebagai penerima data. Untuk pin XCK tidak digunakan karena seluruh *baud rate* yang digunakan telah diatur oleh masing – masing perangkat. *Baud rate* pada ATMEGA8 diatur pada register UBBR dengan Persamaan berikut.

$$UBBR = \frac{f_{osc}}{16 \times baud \ rate} - 1 \quad (2.3)$$

Tabel 2.7 Contoh nilai *error* dengan 2 macam jenis *clock* [2]

Baud rate (bps)	$F_{osc} = 8\text{MHz}$				$F_{osc} = 11.05920\text{MHz}$			
	U2X = 0		U2X = 1		U2X = 0		U2X = 1	
	UBBR	Error	UBBR	Error	UBBR	Error	UBBR	Error
2400	207	0.2%	416	-0.1%	287	0.0%	575	0.0%
4800	103	0.2%	207	0.2%	143	0.0%	287	0.0%
9600	51	0.2%	103	0.2%	71	0.0%	143	0.0%
14.4k	34	-0.8%	68	0.6%	47	0.0%	95	0.0%
19.2k	25	0.2%	51	0.2%	35	0.0%	71	0.0%
28.8k	16	2.1%	34	-0.8%	23	0.0%	47	0.0%
38.4k	12	0.2%	25	0.2%	17	0.0%	35	0.0%
57.6k	8	-3.5%	16	2.1%	11	0.0%	23	0.0%
76.8k	6	-7.0%	12	0.2%	8	0.0%	17	0.0%
115.2k	3	8.5%	8	-3.5%	5	0.0%	11	0.0%
230.4k	1	8.5%	3	8.5%	2	0.0%	5	0.0%
250k	1	0.0%	3	0.0%	2	-7.8%	5	-7.8%
500k	0	0.0%	1	0.0%	-	-	2	-7.8%
1M	-	-	0	0.0%	-	-	-	-
Max	0.5Mbps		1Mbps		691.2kbps		1.3824Mbps	

F_{osc} , merupakan sumber *clock* dari mikrokontroler yang nilainya antara 1MHz – 16 MHz. *Baud rate*, kecepatan pengiriman data pada komunikasi serial berlangsung. Pada **Tabel 2.7** merupakan contoh nilai UBBR dengan berbagai macam *clock* yang digunakan serta *error* yang diperoleh. Untuk nilai *error* dapat dihitung menggunakan Persamaan berikut.

$$Error[\%] = \left(\frac{\text{Baud Rate}_{\text{closest match}}}{\text{Baud Rate}} - 1 \right) \cdot 100\% \quad (2.4)$$

Error maksudnya *error* dalam pengiriman data pada saat komunikasi berlangsung, *error* yang dianjurkan yaitu 0 sampai 0,5%.

2.3 Sensor

- Sensor *Flex*

Flex sensor merupakan sensor *resistance* yang sangat peka terhadap perubahan *bending* dari dirinya. Jadi pada sensor ini nilai resistansinya akan berubah pada saat sensor ditekuk dengan sudut tertentu. Sensor ini diaplikasi untuk sensing gerakan tiap ruas jari. Bahan dari sensor *flex* terdiri dari konduktif device yang memiliki resistansi disetiap elemennya. Perubahan nilai resistansi terjadi akibat dari pergeseran molekul bahan konduktif yang berupa resistansi pada saat sensor menerima *bending* pada bagian sensor.

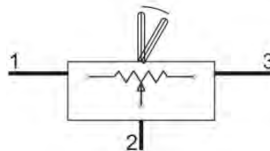


Gambar 2.6 Sensor flex [3]

Spesifikasi dari *Sensor flex* ini memiliki resistansi pada posisi *flat* yaitu sebesar $10 \text{ kohm} \pm 30\%$, perubahan resistansi pada saat *bending* 60 kohm sampai 110 kohm dan daya maksimal $0,5 \text{ Watt}$ untuk penggunaan terus menerus. Aplikasi dari sensor ini adalah untuk game kontroler, bending detector, sensor pada gloves, dan lain sebagainya.

- Sensor Potensiometer

Potensiometer merupakan komponen resistif yang memiliki 3 kaki dengan perubahan dari posisi mekanik. Bentuk dari sensor ini sama dengan variabel resistor yang salah satu kakinya digunakan untuk keluaran dari nilai resistansi.



Gambar 2.7 Simbol Potesiometer



Gambar 2.8 Potensiometer *joystick* [6]

Sensor potensiometer ini sistem kerjanya adalah seperti *wiper* dengan posisi awal berada ditengah. Kaki kedua merupakan nilai keluaran dari sensor potensiometer yang besar nilainya tergantung dari perubahan dari *wiper*. Kaki-kaki dari potensiometer tersebut memiliki nilai resistansi apabila menghubungkannya dengan dua buah kaki saja. Misal 1 dengan 2 atau 3 dengan 2. Untuk potensiometer jenis *wiper* pada umumnya bernilai sebesar $\pm 10k\Omega$, dengan posisi tengah $\pm 5k\Omega$. Derajat perubahan total adalah sebesar 60° dengan posisi tengah sebagai 0° sehingga perubahannya adalah maju 30° dan mundur 30° . Aplikasi dari sensor potensiometer ini adalah sebagai *joy stick* untuk kontroler *game*.

2.4 Bluetooth HC05 dan HC06

Bluetooth merupakan perangkat tanpa kabel yang dapat melakukan komunikasi antara satu sama lainnya. Perangkat ini beroperasi pada frekuensi *bandwidth* 2,4GHz. *Bluetooth* sendiri sejarahnya diambil dari nama raja pada akhir abad ke sepuluh, yaitu Harald Blatand yang sekaligus dijuluki sebagai Harald *Bluetooth* oleh bangsa inggris. Ia merupakan raja dari Negara Denmark yang telah berhasil menyatukan suku suku yang pada awalnya sering berselisih, termasuk suku yang berada diwilayah yang sekarang kita kenal dengan Negara Norwegia dan Swedia. Negara yang menemukan *Bluetooth* ini pun terletak di wilayah Scania Swedia dan sekaligus wilayah kekuasaan dari Harald. Kemampuan dari seorang pemimpin Harald ini dalam menghubungkan berbagai wilayah sama halnya teknologi yang dapat menghubungkan perangkat seperti komputer dan telpon gengam yang ada pada saat ini. Kemudian untuk logo *Bluetooth* sendiri yang sudah universal diambil dari penggabungan dua huruf jerman yang analog dengan huruf H dan B (merupakan singkatan dari Harald *Bluetooth*), yaitu simbol Hagall dan Blatand .



Gambar 2.9 logo *Bluetooth*

Aplikasi dari *Bluetooth* ini digunakan pada perangkat elektronik seperti komputer, telepon genggam, kamera, printer, dan lain sebagainya. Adapun kelebihan dan kekurangan dari *Bluetooth* ini sendiri antara lain. Kelebihan :

- a) Dapat menembus berbagai media seperti dinding, kotak, dan lain sebagainya.
- b) Komunikasi tanpa kabel.
- c) Dapat disinkronkan dengan perangkat yang sama – sama menggunakan *Bluetooth*.

Kekurangan :

- a) Frekuensi yang digunakan sama dengan gelombang pada wifi.
- b) Apabila terdapat banyak koneksi antar *Bluetooth* dalam satu ruangan maka akan sering terjadi gangguan pada saat melakukan komunikasi.
- c) Jarak yang terbatas yaitu sekitar 10 meter.

Kemudian perangkat *Bluetooth* yang digunakan ada dua jenis yaitu sebagai perangkat *Master* dan perangkat *Slave*. Untuk perangkat *Master* digunakan *Bluetooth* dengan seri HC05, perangkat ini dapat difungsikan dengan mode *Slave* penerima data atau sebagai *Master* yang memberi perintah. Sedangkan yang lainnya menggunakan seri HC06 yang telah ditetapkan sebagai *Slave* dan tidak dapat diganti fungsi menjadi *Master*.

2.4.1 *Bluetooth HC05*

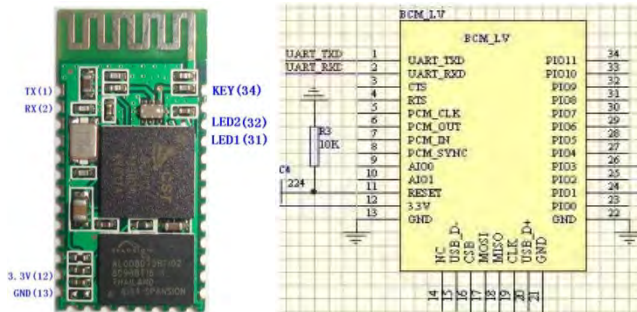
Bluetooth HC05 merupakan *embedded Bluetooth serial communication module*. Untuk *Bluetooth* jenis ini ada dua mode operasi yaitu mode bekerja dengan *inputan* atau diperintah dan mode kerja otomatis terhubung. Maksudnya yaitu untuk *Bluetooth* seri ini akan terkoneksi dengan perangkat lain sesuai dengan perintah saat dia akan melakukan komunikasi dengan perangkat lain, sedangkan mode kerja otomatis pada saat catu daya diaktifkan maka *Bluetooth* akan dengan sendirinya mencari perangkat *Bluetooth* lain untuk melakukan komunikasi.



Gambar 2.10 Modul *Bluetooth* HC05 dengan adapter [4]

HC05 ini bisa digunakan sebagai *Master* yang mengirimkan perintah kepada *Slave*. Ada tiga fungsi untuk mode otomatis yang dapat diseting pada HC05 ini, yaitu bisa dijadikan *Master*, *Slave*, dan *loopback*. Pada saat mode kerja otomatis ini, komunikasi *Bluetooth* mode pengiriman data akan otomatis pada posisi yang telah diseting terakhir kalinya. Contohnya modul diatur sebagai *Master*, maka setingan terakhir ini yang digunakan sebagai setingan *default Bluetooth*. Dan apabila menggunakan mode kerja kontrol, maka akan pengguna harus mengirimkan perintah *AT command* kepada *Bluetooth* untuk mengatur parameter dan mengirimkan perintah kontrol. Untuk mengatur parameter mode kerja pada *Bluetooth* ini dapat diatur melalui modul PIN (PIO11) *input level*. Berikut merupakan deskripsi fitur – fitur dari HC05 :

- a) *Wireless transreceiver*, dengan sensitifitas (bit error rate) sampai -80dBm dan perubahan daya keluaran berkisar antara -4dBm ampai +6dBm.
- b) Memiliki sebuah modul EDR, dengan perubahan modulasi 2Mbps – 3Mbps.
- c) Frekuensi 2,4GHz, tanpa menggunakan antenna.
- d) Memiliki *flash* sebesar 8Mbit
- e) Tegangan kerja 3,1V – 4,2V, dengan arus pada saat *pairing* 30 – 40mA.
- f) Arus pada saat komunikasi sebesar 8mA.
- g) Memiliki port HCI standar UART atau USB
- h) Ukuran yang sangat kecil (27mm x 13mm x 2mm).
- i) Suhu operasi dari *Bluetooth* antar -25°C sampai +75°C.
- j) Gelombang interferensi sebesar 2,4GHz dengan *power emitting* sebesar 3dBm.



Gambar 2.11 Modul *Bluetooth* HC05 [4]

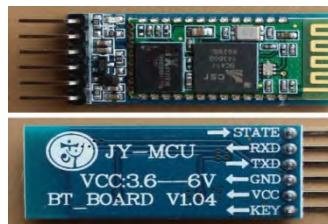
Berikut merupakan keterangan dari pin – pin pada **Gambar 2.11**.

Tabel 2.8 Keterangan pin – pin dari modul *Bluetooth* HC05 [4]

Nama Pin	No. Pin	Jenis pin	Keterangan
UART_TX	1	CMOS output, tri-stable dengan pelemahan pull-down dalam	UART Data output
UART_RX	2	CMOS input, dengan pelemahan pull-down dalam	UART Data input
UART_CTS	3	CMOS input, dengan pelemahan pull-down dalam	Data UART telah terkirim, Aktif low
UART_RTS	4	CMOS output, tri-stable dengan pelemahan pull-down dalam	Perintah UART untuk mengirim data, aktif low
PCM_CLK	5	Bi-Directional	
PCM_OUT	6	CMOS output	
PCM_IN	7	CMOS input	
PCM_SYNC	8	Bi-Directional	
AIO0	9	Bi-Directional	Programmable input/output
AIO1	10	Bi-Directional	Programmable input/output
RESETB	11	CMOS input, dengan pelemahan pull-down dalam	
VCC	12	3.3V	
GND	13	VSS	Ground pot
1.8V	14	VDD	Tersedia catu daya 1.8V (+) dengan regulator dalam dengan keluaran 1,7 – 1,9V
USB -	15	Bi-Directional	
SPI_CSB	16	CMOS input, dengan pelemahan pull-down dalam	
SPI_MOSI	17	CMOS input, dengan	

		pelemahan <i>pull-down</i> dalam	
SPI_MISO	18	CMOS <i>input</i> , dengan pelemahan <i>pull-down</i> dalam	
SPI_CLK	19	CMOS <i>input</i> , dengan pelemahan <i>pull-down</i> dalam	
USB +	20	Bi-Directional	
GND	21	VSS	<i>Ground pot</i>
GND	22	VSS	<i>Ground pot</i>
PIO0	23	Bi-Directional RX EN	<i>Programmable input/output</i> , kontrol untuk LNA (jika dibutuhkan)
PIO1	24	Bi-Directional TX EN	<i>Programmable input/output</i> , kontrol untuk PA (jika dibutuhkan)
PIO2	25	Bi-Directional	<i>Programmable input/output</i>
PIO3	26	Bi-Directional	<i>Programmable input/output</i>
PIO4	27	Bi-Directional	<i>Programmable input/output</i>
PIO5	28	Bi-Directional	<i>Programmable input/output</i>
PIO6	29	Bi-Directional	<i>Programmable input/output</i>
PIO7	30	Bi-Directional	<i>Programmable input/output</i>
PIO8	31	Bi-Directional	<i>Programmable input/output</i>
PIO9	32	Bi-Directional	<i>Programmable input/output</i>
PIO10	33	Bi-Directional	<i>Programmable input/output</i>
PIO11	34	Bi-Directional	<i>Programmable input/output</i>

Prosedur untuk melakukan antarmuka dengan HC05 menggunakan AT *command* adalah sebagai berikut. Proses ini dibantu dengan modul USB to TTL yang digunakan untuk antarmuka *Bluetooth* dengan komputer untuk melakukan pengaturan mode pada HC05.



Gambar 2.12 Pin Keluaran Modul HC05 [4]

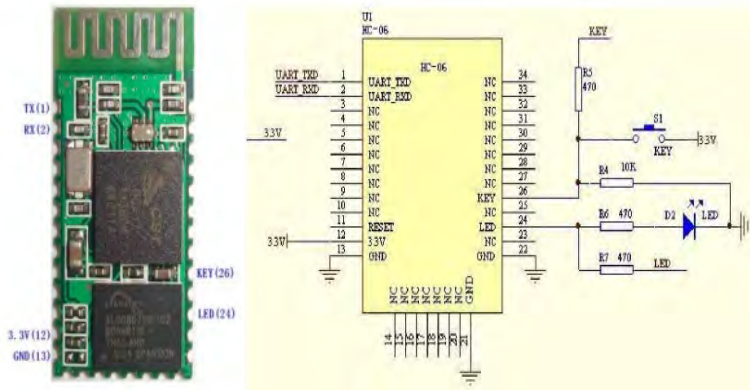


Gambar 2.13 Pin Keluaran Modul USB to TTL

Pemasangan kabel untuk antarmuka HC05 dengan yaitu dengan komunikasi serial dari Tx USB to TTL pada **Gambar 2.13** disambungkan dengan Rx pada HC05 pada **Gambar 2.12** dan Rx USB to TTL disambungkan dengan Tx HC05. Sedangkan untuk catu daya dari HC05 sebesar 3,3V diambil dari 3,3V USB to TTL. Perintah AT *command* dikirimkan untuk memilih mode sebagai *Master* atau *Slave* dengan komunikasi serial. Untuk perintah pengiriman menggunakan “AT+ROLE=1\r\n” role+1 mengatur bahwa HC05 digunakan sebagai *Master*,\r\n merupakan karakter wajib yang perlu ditambahkan pada saat pengiriman AT *command*. Selain itu pada proses ini, juga diatur *baud rate Bluetooth* sesuai dengan perangkat yang akan dikoneksikan. Pada saat *pairing* atau dua buah modul *Bluetooth* saling berkomunikasi (*pairing*) pada HC05 pin LED akan mengeluarkan logika *high*, hal ini akan memudahkan pengguna dalam mengetahui apakah *pairing* antar *Bluetooth* sudah selesai atau masih sedang mencari perangkat. Untuk mengetahuinya pin LED ini digunakan sebagai *input* pada mikroprosesor agar sistem dapat mengetahui *pairing* sudah selesai dan siap melakukan komunikasi. Selain itu ada indikator led pada modul yang akan memperlihatkan aktifitas saat sebelum tersambung dan sudah tersambung. Akan tetapi indikator ini tidak dapat dijadikan acuan untuk mikroprosesor sebagai penentu apakah *pairing* sudah terjadi atau belum. Sehingga akan menyulitkan komunikasi antar *Bluetooth*.

2.4.2 Bluetooth HC06

Modul *Bluetooth* HC06 merupakan perangkat *Bluetooth* yang memiliki satu jenis fungsi sebagai *Master* atau sebagai *Slave* dan mode ini tidak dapat dirubah. Untuk seri HC06 *Master* menggunakan seri HC06M sedangkan *Slave* menggunakan seri HC06S. Disini akan dibahas tentang HC06S yang akan digunakan sebagai *Slave* dan *Master*nya adalah HC05. Variabel dari HC06 yang dapat diganti adalah



Gambar 2.14 Modul HC06 (kiri) dan rangkaian dari modul HC06 (kanan) [5]

setingan *baudrate* untuk kecepatan komunikasi antar *Bluetooth*. Memasukan nilai *baudrate* pada HC06 menggunakan *AT Command*. *Baudrate* yang dapat digunakan yaitu mulai dari 1200 bps sampai 1,3Mbps. **Gambar 2.14** merupakan gambar modul dan rangkaian dari HC06. Penjelasan dari PIN yang terdapat pada modul akan dijelaskan pada **Tabel 2.9**.

Tabel 2.9 Keterangan PIN modul HC06

PIN	Keterangan
PIN1	UART_TXD, TTL/CMOS, UART Data output.
PIN2	UART_RXD, TTL/COMS, UART Data input.
PIN11	RESET, Pin Reset dari modul yang aktif low, pada saat digunakan pin Reset tanpa tidak disambungkan tidak masalah.
PIN12	VCC, merupakan pin catu daya, tegangan standarnya adalah 3.3V, dan dapat aktif pada level tegangan antara 3.0 sampai 4.2V.
PIN13	GND
PIN22	GND
PIN24	Led indikator untuk <i>pairing</i> modul <i>Bluetooth</i> .
PIN26	Digunakan untuk perangkat HC06M(Master), yang berisikan informasi kosong setelah <i>pairing</i> dengan perangkat lain. Setelah dikosongkan Master akan mencari secara acak, dan menyimpan ID dari Slave yang ditemukan. Kemudian, pada saat catu daya dinyalakan perangkat Master hanya akan mencari alamat dari Slave tadi yang sudah terdeteksi.



Gambar 2.15 Modul HC06S [5]

Modul HC06S ini posisi *default*nya adalah mode *Slave*, *baud rate* sebesar 9600, N, 8, 1. Pincode 1234. Mengecek modul HC06S ini dengan menggunakan tambahan perangkat USB to TTL untuk antarmuka *Bluetooth* dengan komputer. Pemasangan kabel untuk antarmuka HC06S dengan yaitu dengan komunikasi serial dari Tx USB to TTL disambungkan dengan Rx pada HC06S dan Rx USB to TTL disambungkan dengan Tx HC06S. Sedangkan untuk catu daya dari HC05 sebesar 3,3V diambil dari 3,3V USB to TTL. Lalu mencoba apakah *Bluetooth* bisa digunakan atau tidak dengan mengirimkan karakter “AT + enter” apabila balasan “OK”, Maka modul HC06 bisa digunakan. Selanjutnya mengganti parameter *default* dengan cara : mengirim perintah “AT+BAUD1” dan menerima balasan “OK1200”. Pilihan *baud rate* yang bisa digunakan adalah :

Tabel 2.10 Pilihan *baud rate* [5]

Pilihan	<i>Baud rate</i> (bps)
1	1200
2	2400
3	4800
4	9600
5	19200
6	38400
7	57600
8	115200
9	230400
A	460800
B	921600
C	1382400

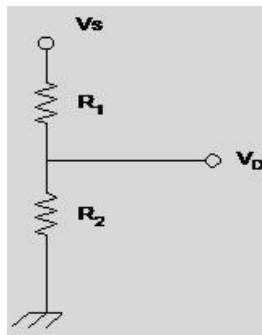
Baud rate yang diatur tidak akan berubah meskipun catu daya dimatikan. Lalu mengubah nama *Bluetooth* dengan mengirimkan “AT+NAMEdevicename” lalu akan mendapat balasan “OKname” dan nama ini akan otomatis tersimpan meskipun catu daya dimatikan. Mengubah *password* dengan perintah “AT+PINxxxx” dan menerima balasan “OKsetpin” *password* akan otomatis tersimpan meskipun catu daya dimatikan.

2.5 Voltage divider

Voltage divider atau rangkaian pembagi tegangan yang dapat digunakan untuk mengkonversi nilai berupa nilai resistansi menjadi nilai tegangan. Dan rangkaian pembagi tegangan ini sangat banyak digunakan dalam mengkonversi nilai sensor menjadi keluaran tegangan seperti photo diode, *flex sensor*, menurunkan level tegangan dan lain sebagainya. Berikut merupakan salah satu contoh dari sebuah rangkaian yang menggunakan metode *voltage divider*.

Syarat dari pengkonversian adalah dengan menggunakan sumber input berupa tegangan dan dua buah resistansi seperti yang terlihat pada **Gambar 2.15** dibawah, dan diberi catu daya masukan (V_s) dengan keluaran tegangan (V_d) diambil dari titik tengah diantara dua buah resistor yang dipasang seri. Dengan demikian keluaran dari rangkaian akan berupa nilai tegangan.

$$V_o = \frac{R_2}{R_1 + R_2} \times V_s \quad (2.5)$$



Gambar 2.16 Rangkaian Pembagi Tegangan

2.6 Op-Amp yang digunakan sebagai buffer

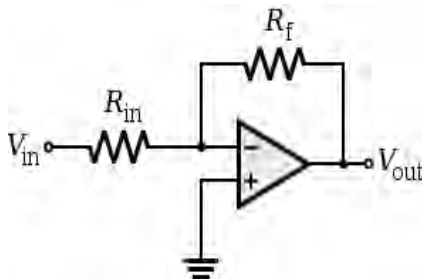
Op-amp merupakan singkatan dari operational amplifier. Op-amp sendiri dapat dioperasikan menjadi dua yaitu, op-amp digunakan sebagai penguat atau pembanding. Kedua ciri ini dapat dilihat dari konfigurasi pemasangan pin dari rangkaian op-amp itu sendiri. Op-amp sebagai penguat pada umumnya bagian dari kaki negative mendapatkan umpan balik dari keluaran op-amp itu sendiri dengan tambahan resistansi pada kaki keluaran dan kaki negative sebagai penghubung pada untai mode *close loop*. Sebaliknya jika op-amp digunakan sebagai pembanding, kaki keluaran dari op-amp akan diumpan balikkan menuju kaki positif. Macam – macam rangkaian pada op-amp jenis penguat:

1. Inverting Amplifier
2. Non-Inverting Amplifier
3. Adder
4. Differential Amplifier
5. Voltage Follower (*Buffer*)

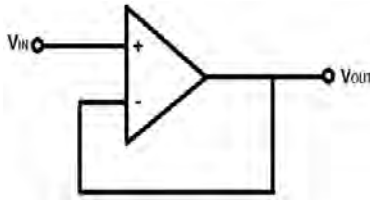
Rangkaian inverting amplifier merupakan rangkaian loop tertutup dengan penguatan sebesar nilai R_f dibagi dengan nilai resistansi input dari op-amp. Untuk lebih jelasnya dapat dilihat pada **Gambar 2.17**. berikut ini.

$$V_{out} = -\frac{R_f}{R_{in}} \times V_{in} \quad (2.6)$$

$$AV = \frac{V_{out}}{V_{in}} = -\frac{R_f}{R_{in}} \quad (2.7)$$



Gambar 2.17 Inverting Amplifier [7]



Gambar 2.18 Voltage Follower [7]

Rangkaian voltage follower merupakan rangkaian loop tertutup dengan pengutan sebesar satu kali yaitu $V_o = V_{in}$. Berikut merupakan rangkaian dari voltage follower.

$$V_{out} = V_{in} \quad (2.8)$$

$$AV = \frac{V_{out}}{V_{in}} = 1 \quad (2.9)$$

Rangkaian voltage follower (*buffer*) ini digunakan sebagai penyangga dari tegangan. Hal ini dikarenakan impedansi input dari op-amp yang besar sehingga tegangan masukan V_{in} akan selalu dipertahankan oleh op-amp sehingga keluaran tegangan akan selalu sama dengan tegangan masukan dari op-amp dan kejadian ini disebut penyangga.

-halaman ini sengaja dikosongkan

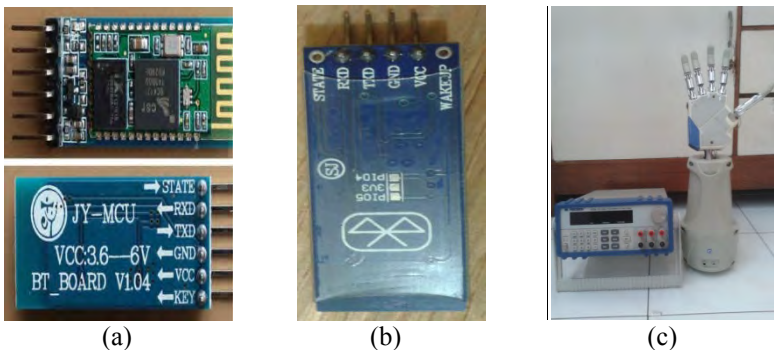
BAB III

PERANCANGAN SISTEM

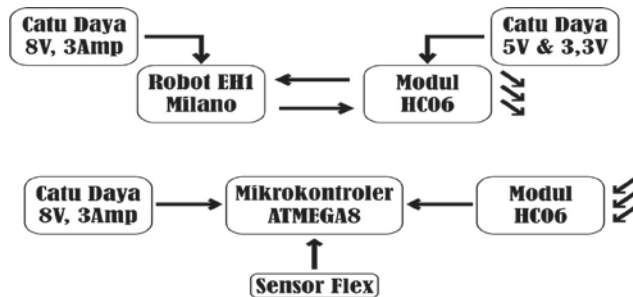
Perancangan sistem kontrol pada robot EH1 Milano terdiri dari beberapa bagian. Bagian – bagian tersebut diantaranya yaitu perancangan elektronis, perancangan mekanik, dan gabungan dari keduanya menjadi realisasi alat. Selain itu pada sub bab ini juga akan membahas tentang blok diagram dari ketiga sistem tersebut. Berikut merupakan uraian dari masing-masing bagian.

3.1 Perancangan Spesifikasi dari Perangkat Keras

Pada sistem kontrol tanpa kabel robot EH1 Milano ini terdapat dua buah perangkat elektronis. Diantaranya yaitu perangkat *Master* yang memberi perintah lalu mengirimkan perintah tersebut dengan komunikasi tanpa kabel, dan *Slave* sebagai perangkat penerima perintah dan sekaligus terhubung dengan perangkat robot EH1 Milano yang merupakan plan yang ingin kita kontrol. Perangkat tanpa kabel menggunakan modul jenis HC. Seri modul HC yang digunakan adalah seri HC05 yang digunakan sebagai *Master* dan seri HC06 digunakan sebagai perangkat *Slave*. Kemudian plan yang akan dikontrol merupakan plan yang berupa tangan robot Prensilia EH1 Milano. Berikut merupakan penjelasannya.



Gambar 3.1 (a) Modul *Bluetooth* HC05, (b) Modul *Bluetooth* HC06, (c) Robot EH1 Milano



Gambar 3.2 Diagram Blok Sistem Secara Umum

3.1.1 Diagram blok

Diagram blok secara keseluruhan dari perancangan sistem elektronis ditunjukkan pada **Gambar 3.2**. dari diagram blok tersebut terbagi menjadi beberapa bagian yaitu, blok pengendali, blok komunikasi, dan blok robot EH1 Milano. Berdasarkan **Gambar 3.2**, blok sensor dan robot EH1 Milano yang merupakan plan yang akan dikontrol merupakan dua sistem yang terpisah dan berhubungan secara *wireless* atau tanpa kabel.

3.1.2 Data Kelistrikan Sistem

Data kelistrikan dari sistem terutama dibagian catu daya terbagi menjadi tiga macam, diantaranya yaitu catu daya robot EH1 Milano, catu daya *Slave*, dan catu daya untuk *Master*. Berikut merupakan uraian dari pencatutan masing – masing sistem.

a) EH1 Milano

Pada sistem pencatutan dari robot EH1 Milano ini merupakan sistem yang sudah ditetapkan dan tidak dapat diubah. Sehingga sistem yang dibuat harus menyesuaikan kepada robot EH1 Milano ini.

Tegangan (DC)	: +8 Volt
Arus	: 3 Amper
Baud rate antar muka UART	: 115200 bps
Level tegangan antar muka	: TTL (0 – 5V)
Resolusi ADC sistem	: 8-bit
Konektor antar muka	: Jack audio tipe stereo 3,5mm
Konektor catu daya	: Jack DC

b) Modul *Slave*

Pada sistem *Slave* ini mengikuti dari yang dibutuhkan untuk bisa melakukan koneksi ke perangkat robot EH1 Milano. Oleh karena sistem ini menggunakan modul *Bluetooth* yang digunakan sebagai antarmuka komunikasi tanpa kabel antara robot dengan perangkat yang memberi instruksi, level tegangan yang digunakan berbeda. Sehingga untuk modul *Slave* ini terdapat dua buah level tegangan yang diuraikan sebagai berikut.

Tegangan (DC)	: +5V dan +3,3V
<i>Baud rate</i> antar muka UART	: 115200 bps
Level tegangan antar muka	: TTL (0 – 5V)
Resolusi ADC sistem	: 8-bit
Konektor antar muka	: <i>Jack audio tipe stereo 3,5mm</i>

c) Modul *Master*

Pada sistem *Master* yaitu menghubungkan perangkat mikrokontroler sebagai otak dari kendali sistem sedangkan data yang dikirim juga menggunakan modul *Bluetooth*. Modul *Bluetooth* ini membutuhkan level tegangan pencatuan yang rendah dan berbeda dengan catu daya sistem pada mikrokontroler sehingga pada modul *Master* ini juga terdapat dua buah catu daya. Berikut merupakan penjelasannya.

Tegangan (DC)	: +5V dan +3,3V
Tegangan referensi ADC	: 2,55V
<i>Baud rate</i> antar muka UART	: 115200 bps
Level tegangan mikrokontroler	: TTL (0 – 5V)
Level tegangan modul HC05	: TTL (0 – 3V)
Resolusi ADC sistem	: 10-bit
Konektor antar muka	: Jalur Tx/Rx
Konektor catu daya	: <i>Jack DC dan JST</i>

d) Catu daya Batre

Batre yang digunakan adalah untuk dijadikan catu daya sistem pada bagian master. Batre yang digunakan adalah tipe litium-ion dari batre *handphone* dengan tegangan sebesar 4,2V. berikut merupakan spesifikasi dari batre litium-ion yang digunakan.

Dimensi (millimeter)	: 43 x 50 x 7
Tegangan Kosong	: 3,7 V
Tegangan Penuh	: 4,18 V
Kapasitas Daya	: 1200 mAh/4,44Wh

3.2 Perancangan Perangkat Keras

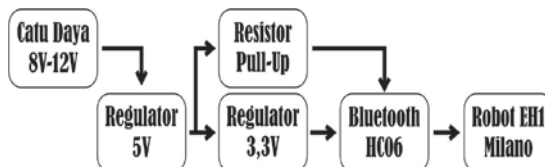
Perancangan perangkat keras (*hardware*) menggunakan *software eagle cadsoft* untuk membuat skematik dan *routing* menjadi *board / layout pcb*. Rancangan *hardware* yang dibuat yaitu modul *Slave* dan modul *Master*.

3.2.1 Modul Slave

Sesuai dengan penjelasan sebelumnya tentang data kelistrikan, yang dibutuhkan pada modul *Slave* agar dapat melakukan *interface* dengan robot EH1 Milano adalah terdapat dua buah catu daya yang salah satunya digunakan untuk menaikkan level tegangan komunikasi Tx si *Bluetooth* ke Rx EH1 Milano. Berikut merupakan penjelasan dari sistem modul *Slave*.

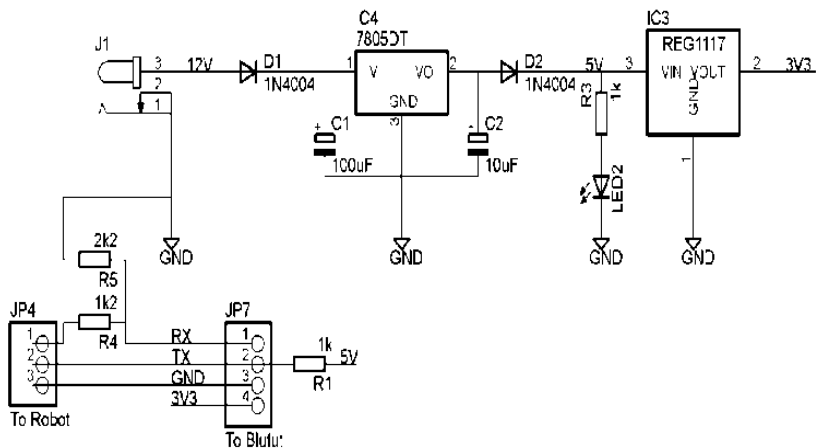
a) Blok Diagram

Dari **Gambar 3.3** dibawah terdapat dua buah catu daya, yang fungsinya yaitu regulator 3,3V untuk *power supplay* dari HC06 yang sebesar 3,3V. Karena level tegangan HC06 sebesar 3,3V ini juga berdampak pada pin Tx/Rx dari HC06 juga sebesar 3,3V. Hal ini menyebabkan beda level tegangan antara robot EH1 Milano dengan sistem komunikasi *Bluetooth*. Oleh karena itu ditambahkan resistor *pull-up* untuk menaikkan level tegangan dari pin Tx agar sama dengan level tegangan robot EH1 Milano sehingga dapat melakukan komunikasi satu sama lainnya. Kemudian pin Tx dari EH1 Milano masuk ke rangkaian pembagi tegangan untuk menurunkan tegangan TTL 5V menjadi 5V agar dapat dibaca oleh modul HC06.



Gambar 3.3 Blok Diagram *Slave*

b) Skematik Rangkaian



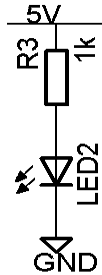
Gambar 3.4 Skematik Rangkaian *Slave* keseluruhan

Dari rangkaian *Slave* diatas sumber tegangan untuk modul *Slave* menggunakan *Jack* DC dengan range 7 – 12V. Tujuan dari design ini adalah karena modul *Slave* berada pada plan yang menggunakan catu dari jack DC maka modul ini pun didesign sama dengan plan.

Tabel 3.1 Tegangan dalam pada led

Warna led	V_{led} (Volt)
Infra Merah	1,6
Merah	1,8 – 2,1
Orange	2,2
Kuning	2,4
Hijau	2,6
Biru	3,0 – 3,5
Putih	3,0 – 3,6
Ultraviolet	3,5

- Rangkaian Indikator Catu



Gambar 3.5 Rangkaian Indikator

Rangkaian ini digunakan untuk memudahkan dalam mengindikasikan pencatutan sistem sudah terpasang dengan baik atau tidak. Apabila led indikator ini menyala pada saat diberi catu daya maka bisa dipastikan pemasangan polaritas dari pemasangan catu daya adalah benar. Untuk nilai tegangan dalam dari led dilihat pada **Tabel 3.1**. Konsumsi daya dari indikator adalah sebesar :

$$\text{daya} = V_s \times i \text{ (watt)} \quad (3.1)$$

Dengan i , sebesar :

$$i_{led} = \frac{(V_s - V_{led})}{R} \quad (3.2)$$

$$i = \frac{(4,2 - 2,6) V}{1000 \text{ Ohm}} = 1,6 \text{ mA} \quad (3.3)$$

$$\text{daya} = 4,2 V \times 1,6 \cdot 10^{-3} \text{ Amp} = 6,72 \text{ mW} \quad (3.4)$$

- Konsumsi Daya Modul HC06 pada Rangkaian *Slave*

Arus dari modul *Bluetooth* HC06 sesuai *data sheet*nya adalah 40mA maksimal. Sehingga dapat dikalkulasikan untuk konsumsi daya dari modul ini adalah sebesar.

$$\text{daya} = V \times i \quad (3.5)$$

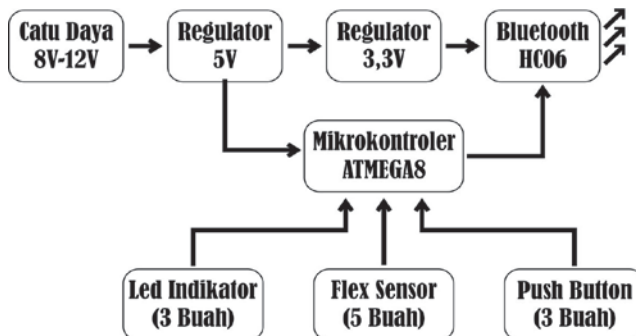
$$\text{daya} = 3,3 V \times 40 \cdot 10^{-3} \text{ Amp} = 0,132 \text{ W} \quad (3.6)$$

3.2.2 Modul Master

Pada modul *Master* ini, tugasnya adalah untuk menerima *input* dari sensor, mengolahnya, dan kemudian mengirimkan data tersebut menuju robot EH1 Milano melalui modul *Bluetooth* HC05. Dari penjelasan sebelumnya tentang data kelistrikan, yang dibutuhkan pada modul *Master* agar dapat mengirimkan data melalui modul HC05. Terdapat perbedaan level tegangan antara ATMEGA8 dengan HC05. Sehingga untuk pin Tx dari mikrokontroler perlu diturunkan menjadi 3V agar dapat melakukan *interface* dengan modul HC05 dan begitu juga sebaliknya untuk pin Rx pada mikrokontroler diberi *pull-up* untuk menaikkan level tegangan dari *Bluetooth* yang sebesar 3V menjadi 5V agar dapat dibaca oleh mikrokontroler. Berikut merupakan penjelasan dari sistem modul *Master*.

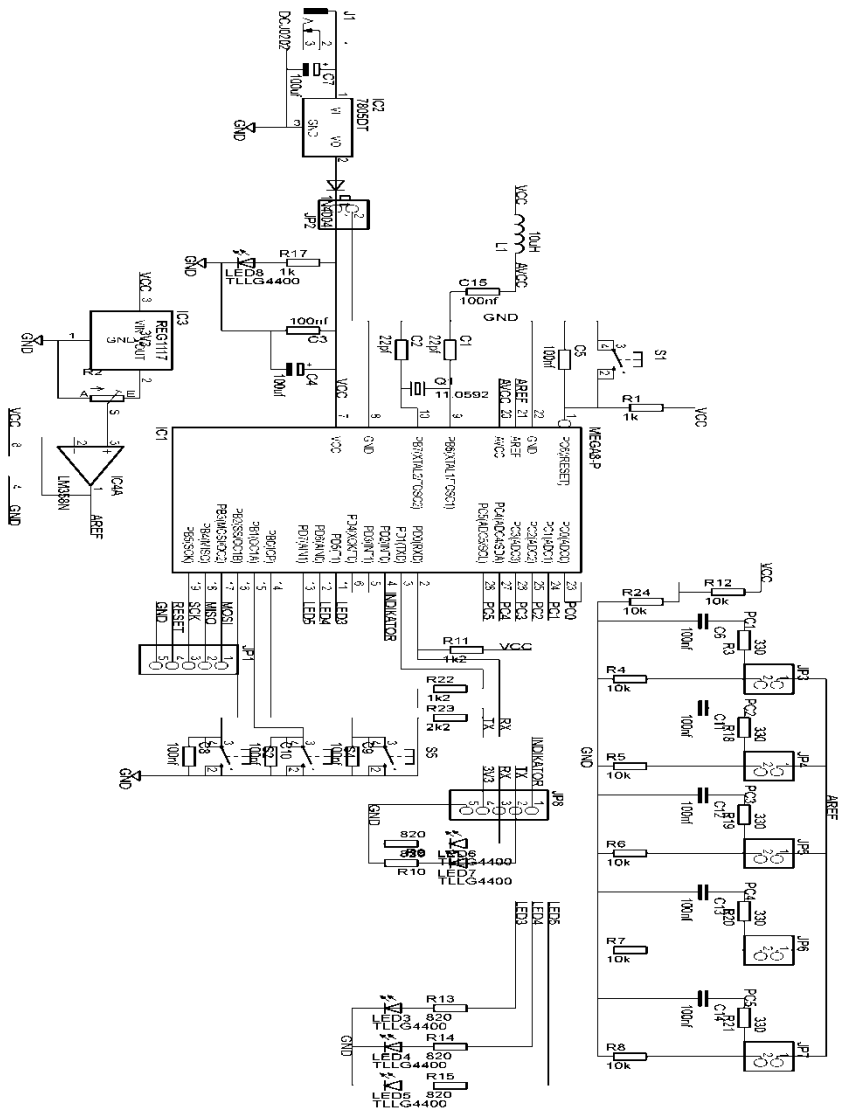
a) Blok Diagram

Blok diagram sistem dari modul *Master* memiliki dua buah catu daya, yang fungsinya yaitu regulator 3,3V untuk *power supply* dari HC05 yang sebesar 3,3V. Karena level tegangan HC05 sebesar 3,3V ini juga berdampak pada pin Tx/Rx dari HC05 juga sebesar 3,3V. Hal ini menyebabkan beda level tegangan antara robot EH1 Milano dengan sistem komunikasi *Bluetooth*. Oleh karena itu ditambahkan rangkaian *Voltage divider* untuk menurunkan level tegangan dari pin Tx ATMEGA8 agar sama dengan level tegangan HC05 dan pin Rx dari mikrokontroler diberi *pull-up* agar level tegangan Tx dari *Bluetooth* menjadi level TTL sebesar 5V sehingga komunikasi antara *Bluetooth* dan mikrokontroler dapat dilakukan.



Gambar 3.6 Blok Diagram *Master*

b) Skematik Rangkaian



Gambar 3.7 Skematik Rangkaian *Master* keseluruhan

Pada rangkaian *Master* terdapat dua macam mode sumber tegangan untuk menyuplai sistem. Pertama adalah dengan menggunakan masukan dari *Jack DC* dengan *range* 7 – 12V. Kedua adalah *bypass* dari keluaran IC regulator 5 volt juga tersedia port batre sebesar 4,2 Volt. Untuk kedua sistem ini dipisahkan dengan menggunakan diode penyearah pada keluaran IC regulator 5V yang mengakibatkan pada waktu mode catu daya batre tidak ada tegangan yang menuju regulator 5 volt karena tertahan oleh diode penyearah. Selain itu, Tujuan dari design ini adalah untuk memudahkan penulis pada waktu menguji sistem agar tidak bergantung kepada catu daya dari batre saja. Karena diode penyearah memotong tegangan yang melewati dirinya sebesar V_d (kurang lebih 0,7 volt). Sehingga sistem ini meskipun tidak tercatu pada batre akan tetapi sumber menggunakan catu daya 7 – 12 volt pun keluarannya akan mendekati tegangan dari batre yang digunakan.

- Rangkaian Indikator Catu

Rangkaian ini digunakan untuk memudahkan dalam mengindikasikan pencatutan sistem sudah terpasang dengan baik atau tidak. Apabila led indikator ini menyala pada saat diberi catu daya maka bisa dipastikan pemasangan polaritas dari pemasangan catu daya adalah benar. Konsumsi daya dari indikator yang terlihat pada **Gambar 3.8** dibawah adalah sebesar :

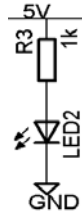
$$daya = V_s \times i \text{ (watt)} \quad (3.7)$$

Dengan i , sebesar :

$$i_{led} = \frac{(V_s - V_{led})}{R} \quad (3.8)$$

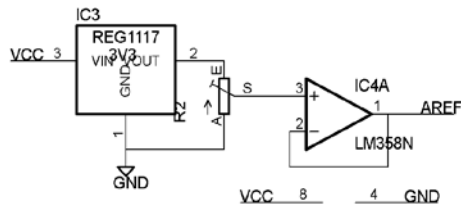
$$i = \frac{(4,2 - 2,6) V}{1000 \text{ Ohm}} = 1,6 mA \quad (3.9)$$

$$daya = 4,2 V \times 1,6 \cdot 10^{-3} \text{ Amp} = 6,72 \text{ mW} \quad (3.10)$$



Gambar 3.8 Rangkaian Indikator dari Catu Batre

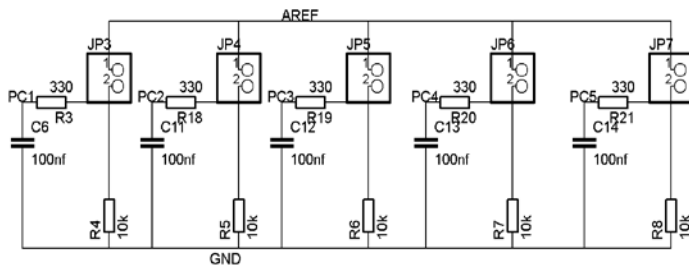
- Rangkaian Aref ADC



Gambar 3.9 Rangkaian Pembagi Tegangan Aref

Rangkaian pembagi tegangan yang diumpangkan menuju rangkaian *buffer* dengan menggunakan ic LM358 dengan penguatan satu kali untuk mengatur besarnya nilai tegangan referensi dari ADC ATMEGA8. Tegangan diatur sebesar 2,048 V, dengan tujuan untuk memudahkan dalam perhitungan pembacaan nilai ADC.

- Rangkaian *Sensor flex*



Gambar 3.10 Rangkaian Pembagi Tegangan Aref

Rangkaian untuk membaca nilai *Sensor flex* adalah rangkaian pembagi tegangan dengan hasil keluaran tegangan sebesar.

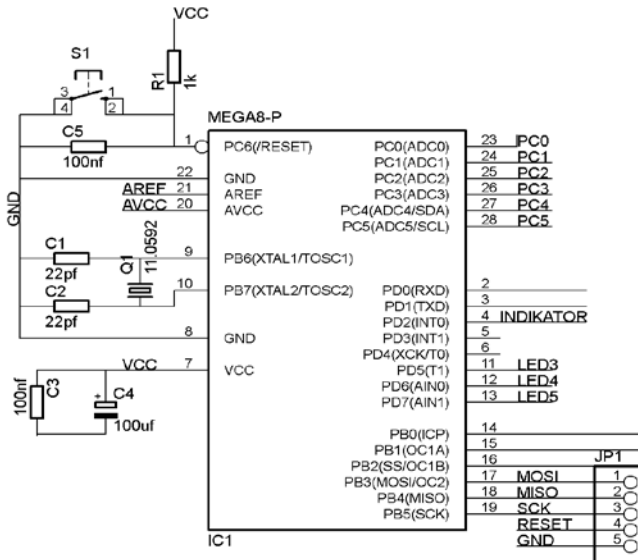
$$V_{Sensor} = \frac{10k}{10k + R_{Flex}} \times Aref \quad (3.11)$$

Keluaran dari rangkaian pembagi tegangan masuk ke rangkaian filter RC, filter RC disini dimanfaatkan untuk waktu tunda (*delay*). Hal ini dilakukan karena tingkat sensitifitas yang tinggi terhadap perubahan *bending* yang diterima oleh sensor. Sehingga dengan adanya waktu tunda ini diharapkan perubahan kecil yang ditimbulkan akibat gerakan jari yang bergetar dapat berkurang. Berikut merupakan rumus untuk menghitung nilai waktu tunda dari rangkaian RC.

$$f(Hz) = \frac{1}{2\pi \cdot R \cdot C} \quad (3.12)$$

$$t(s) = \frac{1}{f} \quad (3.13)$$

- Rangkaian ATMEGA8



Gambar 3.11 Rangkaian ATMEGA8

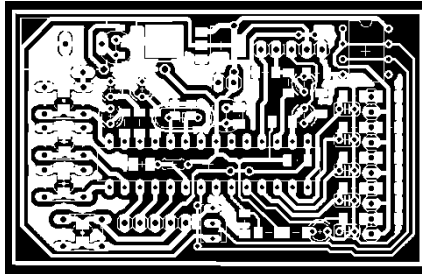
Tabel 3.2 Pin I/O yang digunakan

No. Pin	Pin I/O	Keterangan
2	PD0	Rx
3	PD1	Tx
4	PD2	Indikator <i>feedback</i> dari <i>Bluetooth</i>
11	PD5	Led Indikator 1
12	PD6	Led Indikator 2
13	PD7	Led Indikator 3
14	PB0	<i>Push Button</i> 1
15	PB1	<i>Push Button</i> 2
16	PB2	<i>Push Button</i> 3
17	PB3	MOSI
18	PB4	MISO
19	PB5	SCK
23	PC0	ADC0/Monitoring tegangan batre
24	PC1	ADC1/Sensor 1
25	PC2	ADC2/Sensor 2
26	PC3	ADC3/Sensor 3
27	PC4	ADC4/Sensor 4
28	PC5	ADC5/Sensor 5

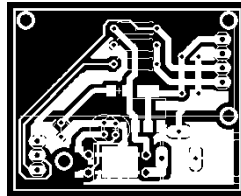
Rangkaian *Master* dengan ATMEGA8 sebagai otak untuk melakukan proses pembacaan nilai sensor dan mengirimkannya ke modul robot EH1 Milano. Pin I/O yang digunakan pada ATMEGA8 akan dipaparkan pada **Tabel 3.2** diatas.

3.2.3 Layout PCB

Rancangan *layout* dan skematik PCB menggunakan *software* Eagle Cadsoft 6.4. Pembuatan *layout* PCB ini dilakukan setelah pembuatan skematik rangkaian. *Layout* PCB yang dirancang hanya menggunakan *single layer* PCB. *Layout* PCB ditunjukkan pada **Gambar 3.12** dan **Gambar 3.13**.



Gambar 3.12 Layout sistem minimum Master ATMEGA8 V.3



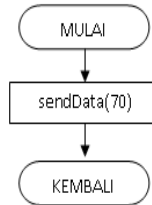
Gambar 3.13 Layout Bluetooth Modul Slave V.3

3.2.4 Perancangan Perangkat Lunak

Pada perancangan perangkat lunak terdiri menggunakan C#AVR. Perancangan perangkat lunak ini antara lain adalah untuk melakukan kalibrasi robot dengan perintah dari mikrokontroler ATMEGA8 yang digunakan dan gerakan robot mengikuti gerakan tangan berdasarkan gerakan yang disensing dengan menggunakan *Sensor flex*.

a) *First calibration*

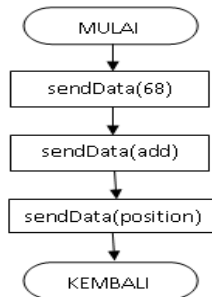
First calibration merupakan perintah yang sudah tersedia pada robot EH1 Milano. Sehingga mikrokontroler hanya perlu mengirimkan perintah 0x70 untuk melakukan *first calibration*, dengan gerakan yang dibuat berupa gerakan satu persatu mulai dari jempol sampai dengan kelingking dan kembali keposisi semula berupa open all, atau semua jari dalam posisi terbuka.



Gambar 3.14 Diagram Blok *First calibration*

b) Perintah *Tracking* Sesuai dengan *Input Sensor*

Pada blok diagram berikut merupakan perintah untuk menggerakkan robot sesuai dengan *inputan* dari sensor. Pengiriman perintah sebanyak 3 *byte*, diantaranya *byte* ke-1 GetFingerPos 0x68 (0b01101000), kemudian *byte* ke-2 mengirimkan alamat dari motor yang hendak digerakkan yaitu antara 0 sampai 5. Lalu yang *byte* ke-3 adalah posisi gerakan dari robot mengikuti *inputan* dari *Sensor flex*.



Gambar 3.15 Diagram Blok *Tracking Sensor*

Tabel 3.3 Pengiriman perintah 3 *byte*

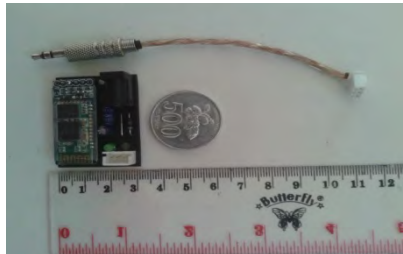
Byte No.	1 st	2 nd								3 rd
Byte Name	SFP	Motor Address								Position
Bin	01000100	0	0	5	4	3	2	1	0	00000000 – 11111111
Hex	0x44	0x00 – 0x20								0x00 – 0xFF

Dec	68	0 - 5	0 - 255
-----	----	-------	---------

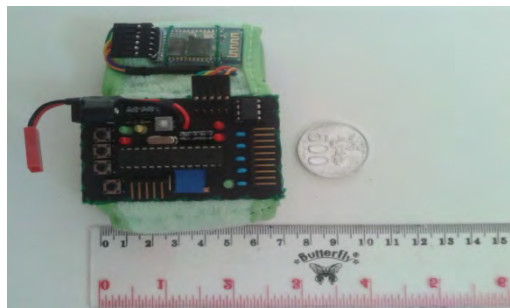
3.3 Realisasi Alat

Setelah semua proses perancangan telah selesai selanjutnya adalah realisasi dari alat yang telah dirancang. Ada tiga bagian dari rancangan yang akan direalisasikan. Diantaranya adalah modul *Slave*, modul sistem minimum *Master*, dan sarung tangan dengan *Sensor flex*.

Untuk realisasi dari modul *Slave* dengan menggunakan PCB jenis fiber dan tembaga jalur dari rangkaian diberi lapisan perak untuk penghantar yang lebih baik. Kemudian untuk komponen sebagian menggunakan komponen ukuran smd. Gambar PCB yang telah dipasang komponen secara lengkap pada **Gambar 3.16**. Pada modul *Slave* ini juga dilengkapi dengan konektor *audio stereo male* untuk akses ke robot EH1 milano. Selanjutnya modul sistem minimum untuk *Master* juga menggunakan bahan yang sama yaitu PCB jenis fiber dan dilapisi dengan perak untuk jalurnya. Kemudian komponen yang digunakan juga sebagian merupakan komponen smd. **Gambar 3.17** merupakan realisasi modul *Master* dengan dilengkapi komponen yang telah disolder pada PCB yang dibuat.



Gambar 3.16 modul *Slave* yang telah di *assembly*



Gambar 3.17 Modul *Master* setelah *diassembly*

Terakhir adalah realisasi sarung tangan versi kedua yang dilengkapi dengan sensor. Sarung tangan merupakan sarung tangan biasa yang digunakan untuk berkendara sebagai salah satu *safety riding tools*. Kemudian sarung tangan dimodifikasi dengan menanamkan *Sensor flex* pada setiap bagian ruas jarinya dengan cara menjahitkan sensor pada bagian dalam dari sarung tangan.



Gambar 3.18 Sarung Tangan Tampak dari Dalam dengan Kain Tambahan Untuk Penanam *Sensor flex*



Gambar 3.19 Sarung Tangan Tampak Dalam dengan *Sensor flex* yang Sudah Ditanam



Gambar 3.20 Sarung Tangan Tampak Dalam



Gambar 3.21 Sarung Tangan telah Dipasang *Sensor flex*



Gambar 3.22 Sarung Tangan Terintegrasi dengan Modul *Master*



Gambar 3.23 Modul *Slave* Terintegrasi dengan Robot EH1 Milano

BAB IV PENGUJIAN DAN ANALISA

4.1 Prosedur dan Proses Pengujian

Pada bagian ini berisikan tentang alat ukur yang digunakan dalam pengujian, prosedur dalam melakukan pengukuran dan pengujian, serta proses dalam melakukan pengukuran dan pengujian.

4.1.1 Alat Ukur yang Digunakan

Proses pengambilan data dari sensor dibutuhkan alat ukur atau *instrumentasi* yang dapat digunakan dalam proses uji coba kelayakan sistem yang telah dibuat. Pada proses pengujian digunakan alat ukur multimeter, osiloskop, meteran, dan stopwatch. Masing – masing alat ukur ini memiliki spesifikasi pengukuran yang berbeda, namun sangat membantu pada proses pengambilan data pengujian.

- a) Multimeter digital buatan Sanwa model CD800a, merupakan alat ukur multi fungsi. Alat ukur yang digunakan dalam proses uji coba adalah untuk pengukuran tegangan DC, resistansi, dan *test point*. Resolusi pengukuran 0,1mV untuk pengukuran tegangan DC dengan akurasi $\pm(0.7\% + 3)$. Resolusi pengukuran 0,1 Ω untuk pengukuran resistansi dengan akurasi $\pm(1.5\% + 5)$ dengan impedansi *input* 11M Ω .



Gambar 4.1 Multimeter Sanwa CD800a



Gambar 4.2 Osiloskop TDS2012B

- b) Osiloskop Textronix TDS2012B, merupakan *instrument* untuk melakukan pengukuran dengan data berupa grafik. Osiloskop ini digunakan untuk melakukan proses pengecekan komunikasi data. Karena osiloskop ini memiliki sampling 1GS/s dan mampu mengukur frekuensi hingga 100MHz.
- c) Meteran (*measuring tape*), merupakan alat ukur jarak yang digunakan dalam proses uji coba adalah untuk pengukuran jarak jangkauan *Bluetooth* HC. Resolusi pengukuran 1 cm.



Gambar 4.3 Meteran



Gambar 4.4 *Stopwatch*

- d) *Stopwatch*, merupakan alat penghitung waktu yang digunakan dalam pengukuran waktu *pairing* dari *Bluetooth* HC.

4.1.2 Prosedur Pengujian

Pengujian yang dilakukan meliputi beberapa bagian. Diantaranya adalah karakteristik sensor tangan dengan Sensor potensiometer, karakteristik sarung tangan dengan *Sensor flex*, range jangkauan *Bluetooth* HC dan efektifitas komunikasi dari *Bluetooth* dalam melakukan komunikasi, pengujian rangkaian *buffer*, dan *test point* pada sistem yang dibuat. Berikut merupakan penjelasan dari prosedur pengujian dari alat yang telah direalisasikan.

- 1) Pengujian karakteristik sensor tangan dengan Sensor potensiometer bertujuan untuk mengetahui tingkat linieritas keluaran dari sensor pada tiap – tiap jari. Pengujian dilakukan dengan menggunakan multimeter digital. Pengujian terdiri dari pengujian perubahan resistansi dari sensor terhadap perubahan sudut jari.
- 2) Pengujian karakteristik sensor tangan dengan *Sensor flex* bertujuan untuk mengetahui tingkat linieritas keluaran dari sensor pada tiap – tiap jari. Pengujian dilakukan dengan menggunakan multimeter digital. Pengujian terdiri dari pengujian perubahan resistansi dari sensor pada saat terjadi perubahan sudut jari.
- 3) Pengujian *range* jangkauan dari *Bluetooth* HC dan efektifitas komunikasi dari *Bluetooth* dalam melakukan komunikasi bertujuan untuk mengetahui seberapa jauh jarak yang dapat dijangkau oleh *Bluetooth* HC dalam melakukan komunikasi. Pengujian dilakukan dengan menggunakan meteran untuk menentukan jarak dalam meter *Bluetooth* dapat melakukan komunikasi. Serta waktu untuk melakukan *pairing* antar *Bluetooth* dihitung dengan menggunakan *stopwatch*. Kemudian dengan jarak pengukuran tersebut juga dilihat

seberapa jauh tingkat keberhasilan dari *Bluetooth* dalam melakukan pengiriman data antar *Bluetooth*.

- 4) Pengujian rangkaian *buffer* bertujuan untuk melihat seberapa jauh rangkaian dalam mempertahankan nilai tegangan dengan beban sensor dan juga sebagai tegangan referensi dari ATMEGA8.
- 5) Pengujian *test point* pada sistem bertujuan untuk menguji kesesuaian sistem yang telah dirancang dengan spesifikasi yang telah ditentukan. Pengujian dilakukan dengan menggunakan *test point* pada multimeter.

4.1.3 Proses Pengujian

Proses pengujian yang dilakukan antara lain adalah sebagai berikut.

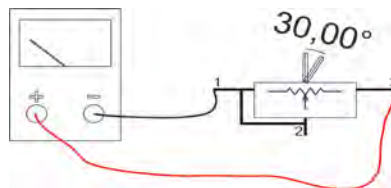
- 1) Pengujian karakteristik sensor tangan dengan Sensor potensiometer.
- 2) Pengujian karakteristik sarung tangan dengan *Sensor flex*.
- 3) Pengujian *range* jangkauan *Bluetooth* HC dan efektifitas komunikasi dari *Bluetooth* dalam melakukan komunikasi.
- 4) Pengujian rangkaian *buffer*.
- 5) Pengujian *test point* pada sistem yang dibuat.

4.2 Hasil Pengujian

Setelah melakukan prosedur pengujian *hardware* pada robot EH1 Milano, diperoleh hasil pengujian berupa data sebagai berikut.

4.2.1 Pengujian Karakteristik Sensor Tangan Dengan Sensor potensiometer

Berikut merupakan blok pengukuran Sensor potensiometer dengan menggunakan alat ukur ohmmeter yang ada pada multimeter Sanwa CD800a.

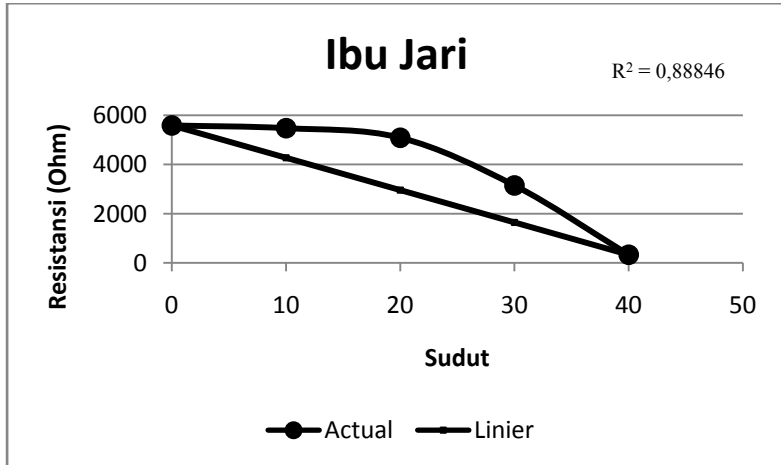


Gambar 4.5 Pengujian Nilai Resistansi Sensor

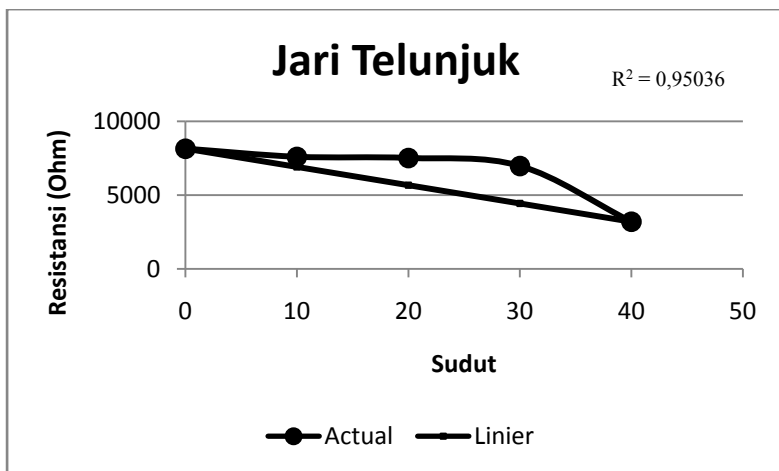
Sensor potensiometer merupakan sensor yang berupa variabel resistor dengan derajat perubahan sebesar 30^0 . Pengukuran sensor dengan keadaan derajat bukaan jari sebesar 0^0 , 10^0 , 20^0 , 30^0 , 40^0 . Pengukuran dilakukan untuk keempat sensor, sehingga linieritas dari sensor dapat diperoleh. Pada **Tabel 4.1** merupakan data hasil pengukuran dari Sensor potensiometer. Dari **Tabel 4.1** dimasukkan kedalam grafik dan diberi *trendline* berupa regresi linier untuk melihat tingkat linieritas dari sensor.

Tabel 4.1 Pengukuran Sensor potensiometer

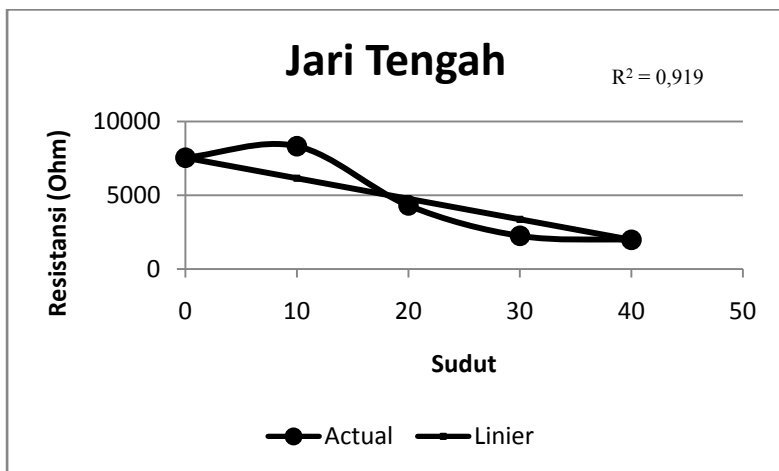
Derajat	Jempol (Ohm)	Telunjuk (Ohm)	Tengah (Ohm)	Manis (Ohm)
0	5590	8150	7550	6920
10	5480	7600	8330	6700
20	5090	7520	4310	4600
30	3150	6980	2250	1410
40	330	3200	1980	1250



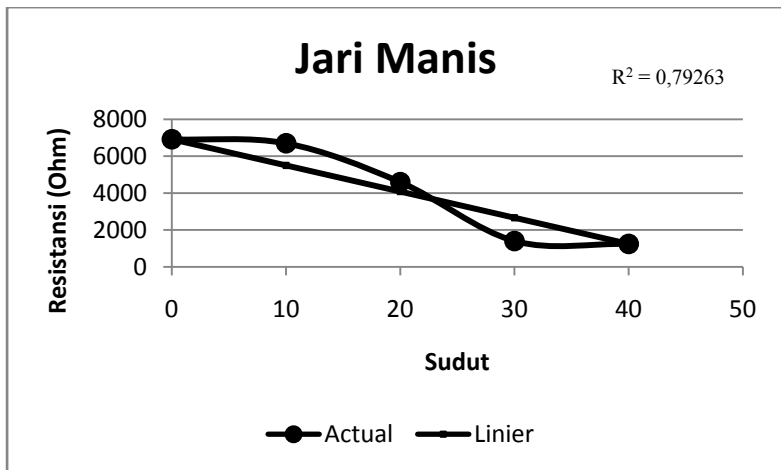
Gambar 4.6 Karakteristik Sensor potensiometer pada Ibu Jari



Gambar 4.7 Karakteristik Sensor potensiometer pada Jari Telunjuk



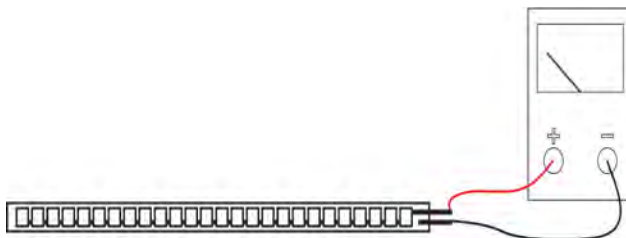
Gambar 4.8 Karakteristik Sensor potensiometer pada Jari Tengah



Gambar 4.9 Karakteristik Sensor potensiometer pada Jari Manis

4.2.2 Pengujian Karakteristik Sensor Tangan Dengan Sensor flex

Berikut merupakan blok pengukuran *Sensor flex* dengan menggunakan alat ukur ohmmeter yang ada pada multimeter Sanwa CD800a. *Sensor flex* merupakan sensor resistansi yang berubah terhadap *bending* yang diterima oleh sensor. Pengukuran sensor dengan keadaan derajat bukaan jari sebesar 0^0 , 10^0 , 20^0 , 30^0 , 40^0 , 50^0 , 60^0 , 70^0 , 80^0 , dan 90^0 . Pengukuran dilakukan untuk kelima sensor, sehingga linieritas dari sensor dapat diperoleh. Pada **Tabel 4.2** berikut merupakan data hasil pengukuran dari *Sensor flex*.

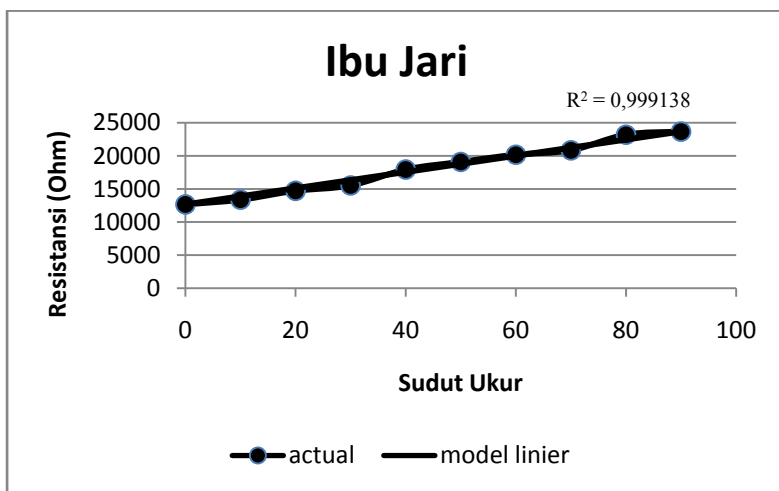


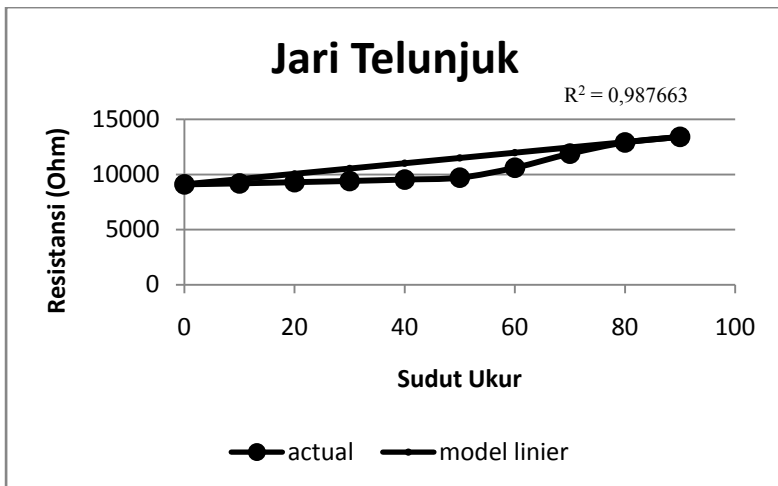
Gambar 4.10 Blok Pengukuran *Sensor flex*

Tabel 4.2 Pengukuran Sensor Flex

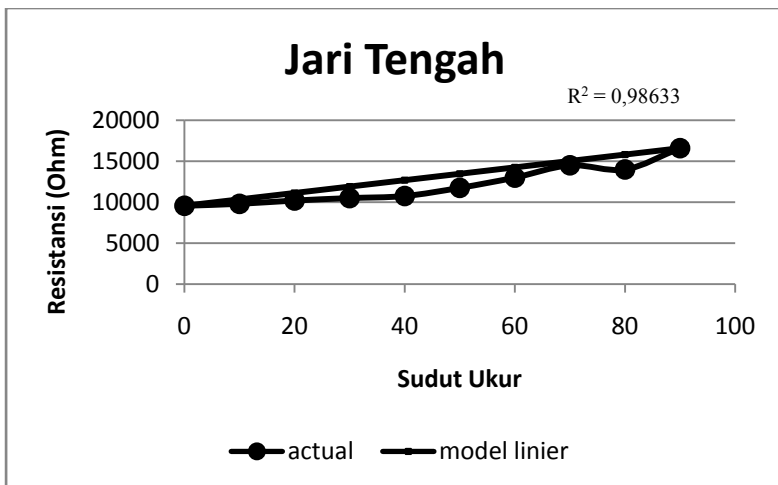
Derajat	Jempol (Ohm)	Telunjuk (Ohm)	Tengah (Ohm)	Manis (Ohm)	Kelingking (Ohm)
0	12650	9100	9560	10700	10000
10	13370	9190	9800	12500	10310
20	14710	9300	10200	13100	10680
30	15520	9410	10500	13560	11040
40	17930	9540	10750	13750	11400
50	19080	9710	11750	14900	12060
60	20170	10610	13000	15400	12620
70	20840	11890	14500	15800	13040
80	23200	12910	14000	16400	13720
90	23680	13420	16600	17250	18350

Dari **Tabel 4.2** diatas dimasukkan kedalam grafik dan diberi *trendline* berupa regresi linier untuk melihat tingkat linieritas dari sensor. Berikut merupakan gambar karakteristik dari sensor.

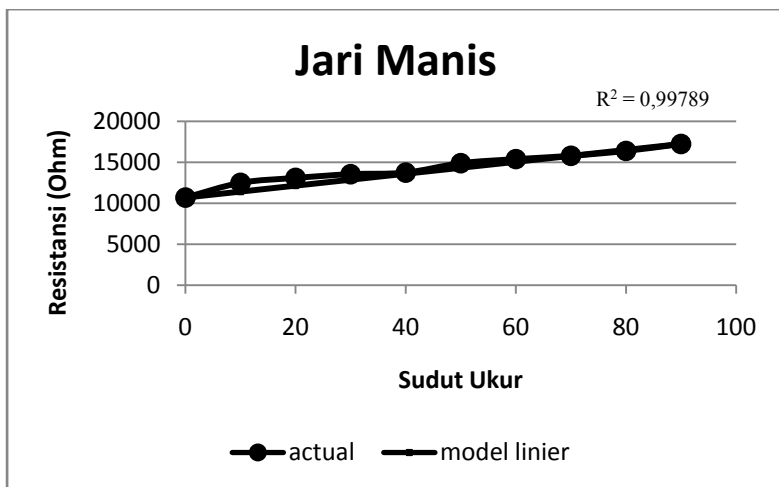
**Gambar 4.11** Karakteristik *Sensor flex* pada Ibu Jari



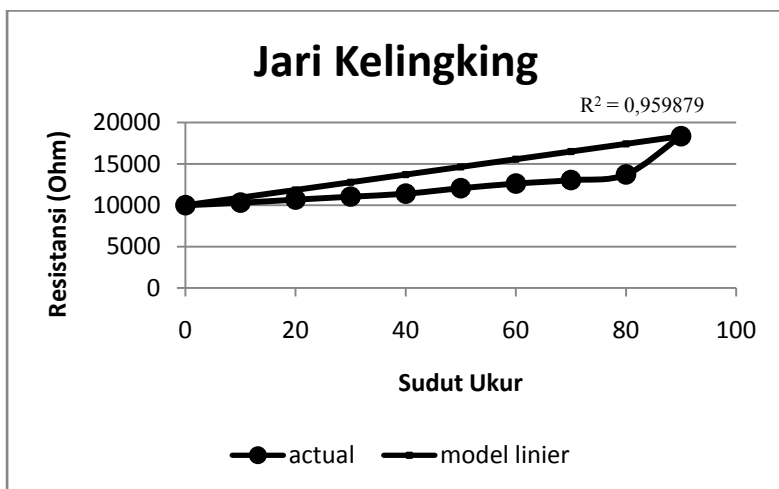
Gambar 4.12 Karakteristik *Sensor flex* pada Jari Telunjuk



Gambar 4.13 Karakteristik *Sensor flex* pada Jari Tengah



Gambar 4.14 Karakteristik *Sensor flex* pada Jari Manis



Gambar 4.15 Karakteristik *Sensor flex* pada Jari Kelingking

4.2.3 Pengujian Range Jangkauan Bluetooth HC dan efektifitas komunikasi dari Bluetooth dalam melakukan komunikasi

Bluetooth HC merupakan perangkat komunikasi tanpa kabel. Komunikasi ke perangkat menggunakan serial dengan data sebanyak 8 bit. Pengujian jarak jangkauan dari modul HC ini adalah untuk menguji kehandalan dari *Bluetooth* dari segi jarak jangkauan yang mampu dicapai oleh modul itu sendiri. Selain itu pada saat pengukuran jarak juga dilakukan komunikasi antar *Bluetooth* yaitu dengan mengirimkan data dari *master* menuju *slave* dan dilihat seberapa *error* yang terjadi dalam proses komunikasi tersebut. Sehingga pengguna dari *Bluetooth* sendiri melakukan komunikasi didalam *range* yang mampu dicapai oleh *Bluetooth* dalam melakukan komunikasi. Pengujian jarak dilakukan dengan dua kondisi, yaitu komunikasi antar *Bluetooth* tanpa halangan dan dengan menggunakan halangan. Waktu estimasi dari *Bluetooth* dalam *pairing* dibatasi sebanyak 5 menit, apabila lewat dari batas tersebut maka *Bluetooth* dinyatakan tidak dapat terhubung atau gagal *pairing*. Berikut merupakan tabel pengujian dari *Bluetooth* HC.



Gambar 4.16 Blok Pengukuran Jangkauan *Bluetooth* HC

Tabel 4.3 Pengujian jarak *Bluetooth* HC kondisi tanpa halangan

No.	Jarak (meter)	Kondisi	<i>Pairing Time</i> (s)	Status
1	1	Tanpa halangan	6,2	<i>Connect</i>
2	2	Tanpa halangan	6,5	<i>Connect</i>
3	3	Tanpa halangan	6,4	<i>Connect</i>
4	4	Tanpa halangan	6,5	<i>Connect</i>
5	5	Tanpa halangan	6,4	<i>Connect</i>
6	10	Tanpa halangan	6,5	<i>Connect</i>
7	15	Tanpa halangan	6,5	<i>Connect</i>

8	20	Tanpa halangan	6,5	<i>Connect</i>
9	25	Tanpa halangan	7	<i>Connect</i>
10	30	Tanpa halangan	7,3	<i>Connect</i>
11	35	Tanpa halangan	8	<i>Connect</i>
12	40	Tanpa halangan	9,2	<i>Connect</i>
13	45	Tanpa halangan	9,8	<i>Connect</i>
14	50	Tanpa halangan	170	<i>Connect</i>
15	>50	Tanpa halangan	300	<i>Not Connected</i>

Tabel 4.4 Pengujian jarak *Bluetooth* HC kondisi dengan halangan

No.	Jarak (meter)	Kondisi	<i>Pairing Time</i> (s)	Status
1	1	Dengan Halangan	6,2	<i>Connect</i>
2	2	Dengan Halangan	6,5	<i>Connect</i>
3	3	Dengan Halangan	6,4	<i>Connect</i>
4	4	Dengan Halangan	6,5	<i>Connect</i>
5	5	Dengan Halangan	6,4	<i>Connect</i>
6	6	Dengan Halangan	6,5	<i>Connect</i>
7	7	Dengan Halangan	6,5	<i>Connect</i>
8	8	Dengan Halangan	6,5	<i>Connect</i>
9	9	Dengan Halangan	7	<i>Connect</i>
10	10	Dengan Halangan	200	<i>Connect</i>
11	>10	Dengan Halangan	300	<i>Not connected</i>

Pengujian efektifitas pengiriman data dari modul *wireless* dengan metode monitoring menggunakan komputer untuk melihat tingkat keberhasilan dari data yang dikirim oleh salah 1 modul menuju modul lainnya. Data yang dikirim merupakan data dari modul *master* menuju *slave*, dengan data yang dikirim berupa data *fast calibration* dengan list program sebagai berikut.

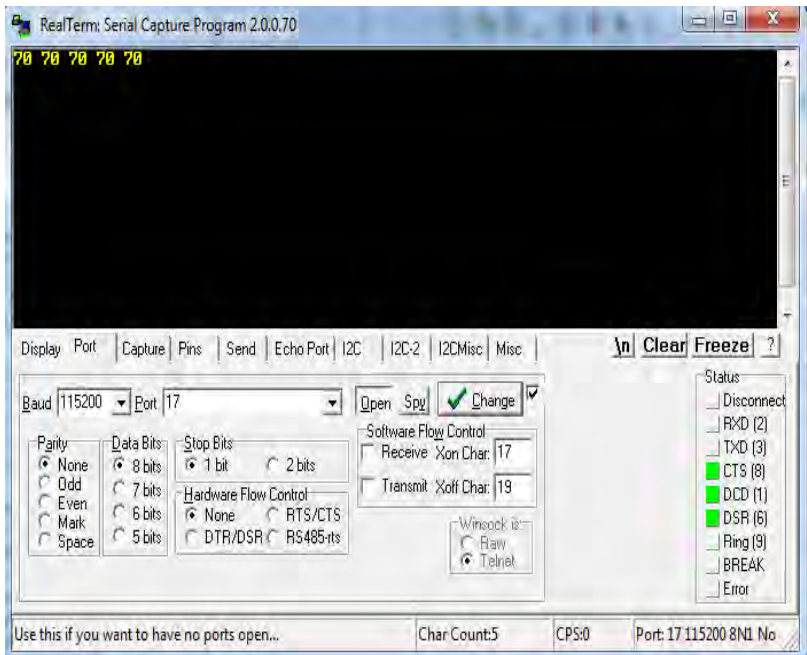
```
#define FastCalb    70
#define fc    !PINB.1
```

```

if(fc)
{
    while(fc);
    sendData(FastCalb);
    PORTD.5=0;
    PORTD.6=1;
    #asm("cli")
}

```

Jadi, setiap tombol pada pin B1 maka mikrokontroler akan mengirimkan data *fast calibration* dengan data 70 yang merupakan bilangan decimal. Kemudian, **Tabel 4.5** dan **Tabel 4.6** berikut merupakan tabel pengujian pengiriman data dari keterangan di atas dengan jarak yang berbeda-beda. Dan data yang diterima terlihat pada **Gambar 4.17** berikut oleh monitoring yang dilakukan pada komputer.



Gambar 4.17. Monitoring penerimaan data dari modul wireless dengan menggunakan RealTime Serial Capture Program.

Tabel 4.5 Pengujian pengiriman data dengan halangan

No.	Jarak (meter)	Status	Uji Coba ke-									
			1		2		3		4		5	
			K	T	K	T	K	T	K	T	K	T
1	1	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
2	2	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
3	3	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
4	4	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
5	5	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
6	6	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
7	7	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
8	8	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
9	9	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
10	10	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
11	>10	<i>Not connected</i>	-	-	-	-	-	-	-	-	-	-

Tabel 4.6 Pengujian pengiriman data tanpa halangan

No.	Jarak (meter)	Status	Uji Coba ke-									
			1		2		3		4		5	
			K	T	K	T	K	T	K	T	K	T
1	1	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
2	2	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
3	3	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
4	4	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
5	5	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
6	10	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
7	15	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70
8	20	<i>Connect</i>	70	70	70	70	70	70	70	70	70	70

9	25	Connect	70	70	70	70	70	70	70	70	70	70
10	30	Connect	70	70	70	70	70	70	70	70	70	70
11	35	Connect	70	70	70	70	70	70	70	70	70	70
12	40	Connect	70	70	70	70	70	70	70	70	70	70
13	45	Connect	70	70	70	70	70	70	70	70	70	70
14	50	Connect	70	70	70	70	70	70	70	70	70	70
15	>50	Not connected	-	-	-	-	-	-	-	-	-	-

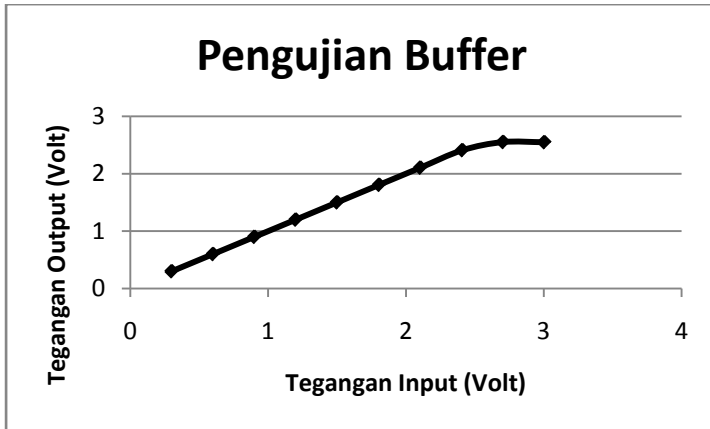
4.2.4 Pengujian Rangkaian Buffer

Rangkaian *buffer* merupakan rangkaian penyangga dengan impedansi masukan yang besar sehingga nilai keluaran dari rangkaian *buffer* ini akan sama dengan nilai tegangan masukan. **Tabel 4.5** berikut merupakan hasil pengukuran nilai tegangan keluaran dan masukan dari rangkaian *buffer*.

Tabel 4.7 Data hasil pengujian rangkaian buffer

No.	V _{in}	V _{out}
1	0,3	0,3
2	0,6	0,6
3	0,9	0,9
4	1,2	1,2
5	1,5	1,501
6	1,805	1,807
7	2,104	2,106
8	2,409	2,411
9	2,705	2,553
10	3,004	2,55

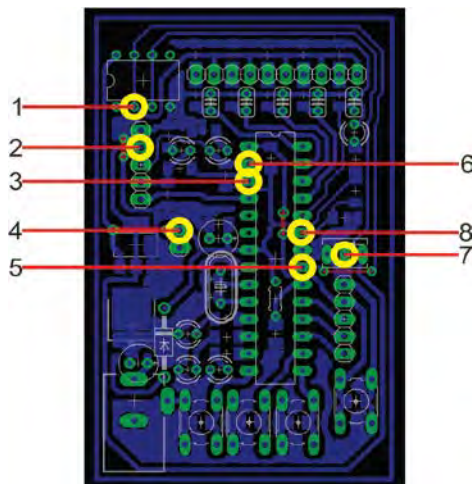
Dari data diatas dijadikan kedalam bentuk gambar grafik seperti pada **Gambar 4.18**.



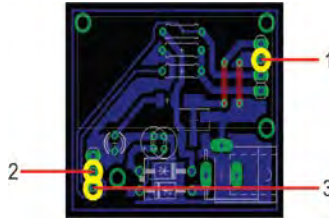
Gambar 4.18. Grafik pengujian rangkaian *buffer*

4.2.5 Pengujian Test point Pada Sistem yang Dibuat

Pengujian *test point* pada sistem yang dibuat ada dua yaitu pengujian pada modul *Master* dan modul *Slave*. Berikut merupakan gambar titik pengujian pada modul *Master* dan modul *Slave*.



Gambar 4.19 Pengujian *test point* pada modul *Master*



Gambar 4.20 Pengujian *test point* pada modul *Slave*

Berikut merupakan tabel hasil pengukuran dari *test point*, pengukuran untuk modul *Master* menggunakan dua pengujian yaitu menggunakan batre dan menggunakan *Power supply* dari luar. *Power supply* yang digunakan adalah tegangan adaptor dengan keluaran DC 12 Volt untuk modul *master* dan 8 Volt untuk modul *slave*.

Tabel 4.8 Pengujian *test point* modul *Master*

No.	<i>Test point</i>	Tegangan (Volt)	
		Mode Batre	<i>Power supply</i>
1	<i>Output buffer / Vref</i>	2,048	2,048
2	Catu daya <i>Bluetooth</i>	3,168	3,239
3	<i>Voltage divider</i> pin Tx	2,383	2,41
4	Vbat	4,04	4,14
5	Avcc	4,04	4,14
6	<i>Pull-up</i> pin Rx	3,99	4,09
7	<i>Input buffer</i>	2,046	2,047
8	Vbat di ADC	2,006	2064

Tabel 4.9 Pengujian *test point* modul *Slave*

No.	<i>Test point</i>	Tegangan (Volt)
1	Catu daya <i>Bluetooth</i>	3,267
2	<i>Pull-up</i> pin Tx	3,047
3	<i>Voltage divider</i> pin Rx	0

Gambar bulatan berwarna kuning merupakan *test point* dari rangkaian baik *Master* maupun *Slave*. Ada delapan macam pengujian *test point* untuk modul *Master* dan tiga macam untuk modul *Slave*. Data hasil pengukuran ditunjukkan pada **Tabel 4.8** dan **Tabel 4.9**. Tegangan pada modul *Slave* sama dengan tegangan pada *Power supply* pada modul *Master* yaitu sebesar 8 Volt. Sedangkan tegangan batre yang digunakan pada modul *Master* sebesar 4,08 Volt.

4.3 Analisa

Setelah melakukan pengujian dan didapatkan data – data hasil pengujian, kemudian dari data – data tersebut diperoleh analisa yang sesuai dengan spesifikasi perancangan dan yang belum sesuai dengan perancangan.

4.3.1 Analisa Karakteristik Sensor Tangan dengan Sensor potensiometer

Perubahan resistansi sensor pada masing – masing jari tidak sama, hal ini disebabkan oleh faktor nilai titik tengah dari sensor yang berbeda-beda. Hal ini terlihat pada grafik yang ditampilkan sebelumnya. Untuk tingkat linieritas dari sensor dapat dihitung dengan menggunakan rumus garis linier berikut ini.

$$(y - y_1) = \frac{(y_2 - y_1)}{(x_2 - x_1)} (x - x_1) \quad (4.1)$$

Dari Persamaan diatas, dimasukkan untuk mendapatkan garis linier dari data sebelumnya. Data hasil perhitungan ditunjukkan oleh **Tabel 4.10** berikut.

Tabel 4.10 Data garis linier Sensor Potensiometer

Derajat	Jempol	Telunjuk	Tengah	Manis
0	5590	8150	7550	6920
10	4275	6912,5	6157,5	5502,5
20	2960	5675	4765	4085
30	1645	4437,5	3372,5	2667,5
40	330	3200	1980	1250

Sehingga dapat dihitung *mean percentage error* dari data Sensor potensiometer dengan rumus berikut.

$$MPE = \left| \frac{100\%}{n} \sum_t^x \frac{f_t - a_t}{a_t} \right| \quad (4.2)$$

Keterangan :

- MPE : *Mean Percentage Error*
 n : Jumlah data
 t : Merupakan titik awal dimulai iterasi
 x : titik berakhirnya iterasi
 f_t : *Data forecast* (data garis linier)
 a_t : *Data actual* (data sensor yang sebenarnya)

Dari rumus tersebut dapat diperoleh nilai *error* dari sensor terhadap garis linier adalah sebesar .

$$MPE_{jempol} = \left| \frac{100\%}{5} \sum_1^5 \frac{f_t - a_t}{a_t} \right| = 22,32\% \quad (4.3)$$

$$MPE_{telunjuk} = \left| \frac{100\%}{5} \sum_1^5 \frac{f_t - a_t}{a_t} \right| = 14\% \quad (4.4)$$

$$MPE_{tengah} = \left| \frac{100\%}{5} \sum_1^5 \frac{f_t - a_t}{a_t} \right| = 6,87\% \quad (4.5)$$

$$MPE_{manis} = \left| \frac{100\%}{5} \sum_1^5 \frac{f_t - a_t}{a_t} \right| = 12,02\% \quad (4.6)$$

Dari data perhitungan diatas terlihat bahwa *error* terbesar adalah 22,32% menunjukkan bahwa tingkat linieritas dari sensor kurang baik. Hal ini terjadi akibat pemasangan mekanik dari sensor pada sarung tangan sehingga sudut perubahan setiap jari dapat berbeda. Selain itu peanrikan sensor menggunakan benang yang tingkat ketegangannya berbeda untuk tiap-tiap sensor. Berikut merupakan konsumsi daya total dari sensor potensiometer.

$$\frac{1}{R_{total}} = \frac{1}{R_{p.sens\ 1}} + \frac{1}{R_{p.sens\ 2}} + \frac{1}{R_{p.sens\ 3}} + \frac{1}{R_{p.sens\ 4}} \quad (4.7)$$

$$\frac{1}{R_{total}} = \frac{1}{10160} + \frac{1}{9320} + \frac{1}{10270} + \frac{1}{9860} \quad (4.8)$$

$$R_{total} = 2472,11 \text{ Ohm} \quad (4.9)$$

$$P_{sensor} = \frac{4,08^2}{2472,11} = 6,73mW \quad (4.10)$$

4.3.2 Analisa Karakteristik Sensor Tangan dengan *Sensor flex*

Perubahan resistansi sensor pada masing – masing jari tidak sama, hal ini disebabkan oleh faktor suhu, kelembaban, dan toleransi nilai resistansi dari sensor. Hal ini terlihat pada grafik yang ditampilkan sebelumnya. Untuk tingkat linieritas dari sensor dapat dihitung dengan menggunakan rumus garis linier berikut ini.

$$(y - y_1) = \frac{(y_2 - y_1)}{(x_2 - x_1)} (x - x_1) \quad (4.11)$$

Dari Persamaan diatas, dimasukkan untuk mendapatkan garis linier dari data sebelumnya. Data hasil perhitungan ditunjukkan oleh **Tabel 4.11** berikut.

Tabel 4.11 Data garis linier *Sensor flex*

Derajat	Jempol	Telunjuk	Tengah	Manis	kelingking
0	12650	9100	9560	10700	10000
10	13875,556	9580	10342,222	11427,778	10927,778
20	15101,111	10060	11124,444	12155,556	11855,556
30	16326,667	10540	11906,667	12883,333	12783,333
40	17552,222	11020	12688,889	13611,111	13711,111
50	18777,778	11500	13471,111	14338,889	14638,889
60	20003,333	11980	14253,333	15066,667	15566,667
70	21228,889	12460	15035,555	15794,445	16494,445
80	22454,444	12940	15817,778	16522,222	17422,222
90	23680	13420	16600	17250	18350

Sehingga dapat dihitung *mean percentage error* dari data Sensor potensiometer dengan rumus berikut.

$$MPE = \left| \frac{100\%}{n} \sum_t^x \frac{f_t - a_t}{a_t} \right| \quad (4.12)$$

Keterangan :

- MPE : *Mean Percentage Error*
 n : Jumlah data
 t : Merupakan titik awal dimulai iterasi
 x : titik berakhirnya iterasi
 f_t : *Data forecast* (data garis linier)
 a_t : *Data actual* (data sensor yang sebenarnya)

Dari rumus tersebut dapat diperoleh nilai *error* dari sensor terhadap garis linier adalah sebesar .

$$MPE_{jempol} = \left| \frac{100\%}{10} \sum_1^{10} \frac{f_t - a_t}{a_t} \right| = 0,58\% \quad (4.13)$$

$$MPE_{telunjuk} = \left| \frac{100\%}{10} \sum_1^{10} \frac{f_t - a_t}{a_t} \right| = 7,63\% \quad (4.14)$$

$$MPE_{tengah} = \left| \frac{100\%}{10} \sum_1^{10} \frac{f_t - a_t}{a_t} \right| = 8,7\% \quad (4.15)$$

$$MPE_{manis} = \left| \frac{100\%}{10} \sum_1^{10} \frac{f_t - a_t}{a_t} \right| = 2,7\% \quad (4.16)$$

$$MPE_{manis} = \left| \frac{100\%}{10} \sum_1^{10} \frac{f_t - a_t}{a_t} \right| = 15,13\% \quad (4.17)$$

Dari data perhitungan diatas terlihat bahwa *error* terbesar adalah 15,13% menunjukkan bahwa tingkat linieritas dari sensor cukup baik. Hal ini karena sensor dapat mengetahui atau memberi nilai setiap adanya perubahan untuk setiap lekukan dari ruas jari. Berikut merupakan konsumsi daya total dari sensor flex.

$$\frac{1}{R_{total}} = \frac{1}{R_{p.sens\ 1}} + \frac{1}{R_{p.sens\ 2}} + \frac{1}{R_{p.sens\ 3}} + \frac{1}{R_{p.sens\ 4}} + \frac{1}{R_{p.sens\ 5}} \quad (4.18)$$

$$\frac{1}{R_{total}} = \frac{1}{23680} + \frac{1}{13420} + \frac{1}{16600} + \frac{1}{17250} + \frac{1}{18350} \quad (4.19)$$

$$\frac{1}{R_{total}} = \frac{1}{23680} + \frac{1}{13420} + \frac{1}{16600} + \frac{1}{17250} + \frac{1}{18350} \quad (4.20)$$

$$R_{total} = 3454,79 \text{ Ohm} \quad (4.21)$$

$$P_{sensor} = \frac{4.08^2}{3454,79} = 4,82mW \quad (4.22)$$

4.3.3 Analisa Range Jarak Bluetooth HC dan efektifitas komunikasi dari Bluetooth dalam melakukan komunikasi.

Jarak yang dapat dijangkau oleh Bluetooth tipe HC tanpa menggunakan penghalang mampu mencapai 50 meter dengan waktu *pairing* yang cukup lama yaitu 2 menit 50 detik. Hal ini terjadi karena faktor jarak dan tempat pengukuran yang dilakukan ditempat terbuka dimana banyak sinyal dari luar seperti wifi, handphone, dan lain sebagainya. Yang dapat mengganggu Bluetooth dalam proses *pairing*. Akan tetapi pada jarak kurang dari 50 meter *pairing* dari Bluetooth normal yaitu kurang dari 10 detik. Sedangkan lebih dari 50 meter Bluetooth sudah tidak mampu lagi melakukan komunikasi atau *pairing*.

Pengujian dengan halangan dilakukan di dua ruangan berbeda yaitu Master di ruang A dan Slave diruang B. Jarak terjauh yang dapat dijangkau oleh Bluetooth adalah 10 meter saja dengan waktu *pairing* 3 menit 20 detik. Lebih dari 10 meter dengan waktu estimasi 5 menit *pairing* tidak dapat dilakukan sehingga dinyatakan Bluetooth dengan jarak lebih dari 10 meter untuk dengan halangan Bluetooth tidak dapat melakukan komunikasi. Untuk jarak kurang dari 10 meter waktu *pairing* normal yaitu dengan waktu kurang dari 10 detik.

Untuk pengujian pengiriman data dengan jarak yang berbeda dilakukan pengujian sebanyak lima kali dan pengujian menggunakan halangan dan tanpa halangan. Dari data yang diperoleh setiap status Bluetooth *connected* maka tingkat keberhasilan pengiriman data antara data yang dikirim sama dengan data yang diterima. Sehingga validitas data menggunakan modul *wireless* dari komunikasi yang dilakukan adalah 100%.

4.3.4 Analisa pengujian rangkaian *buffer*

Tegangan keluaran dari rangkaian *buffer* mampu dipertahankan sebesar tegangan masukannya hingga input sebesar 2,4V. Namun pada tegangan input > 2,4V keluaran dari *buffer* stack pada tegangan 2,55V. hal ini terjadi karena tegangan tegangan keluaran dari op-amp sebagai *buffer* berada dalam keadaan saturasi. Tegangan batre pada saat pengukuran adalah sebesar 4,08V.

4.3.5 Analisa *Test Point* pada Sistem

- Modul *slave*

Tegangan keluaran pada data pengukuran test point untuk modul *slave* dengan catu daya dari adaptor keluaran DC 8V sudah mendekati dari spesifikasi yang dirancang.

$$e_{catu\ daya} = \left| \frac{f_t - a_t}{a_t} \right| \times 100\% \quad (4.23)$$

$$e_{catu\ daya} = \left| \frac{3,3 - 3,267}{3,267} \right| \times 100\% = 1,01\% \quad (4.24)$$

*Error*nya kecil sekali yaitu 1,01% ini menandakan bahwa perancangan sistem sudah mendekati dari nilai dari spesifikasi yang ditentukan.

Kemudian untuk pin Tx yang di *pull-up* nilainya adalah 4,71, akan tetapi 4,71 sudah termasuk kedalam kategori tegangan TTL dan cukup untuk melakukan komunikasi serial dengan level tegangan TTL 0 – 5V. sedangkan untuk pin Rx adalah dari level TTL menuju *Bluetooth* yang level tegangannya adalah 0 – 3V. Pada saat terjadi komunikasi tegangan dari plan sebesar 5V masuk kerangkaian pembagi tegangan untuk menurunkan level tegangan dari plan menjadi 3V agar dapat dibaca oleh *Bluetooth*.

- Modul *master*

Tegangan masukan untuk modul *master* dari adaptor sebesar 12,26 V, sedangkan dari batre sebesar 4,08 Volt. Dari data yang diperoleh terdapat sedikit perbedaan dari spesifikasi yang ditentukan. Adapun *error* tersebut adalah sebesar.

$$e_{Vref} = \left| \frac{2,048 - 2,048}{2,048} \right| \times 100\% = 0\% \quad (4.25)$$

Selain itu, juga terjadi perbedaan pada catu daya input untuk *Bluetooth* dengan menggunakan batre dan adaptor. Berikut merupakan selisih nilai dari perdaan tersebut.

$$e_{Catu\ batre} = \left| \frac{3,3-3,168}{3,168} \right| \times 100\% = 4,17\% \quad (4.26)$$

$$e_{Catu\ adaptor} = \left| \frac{3,3-3,239}{3,239} \right| \times 100\% = 1,88\% \quad (4.27)$$

Untuk *voltage divider* diperoleh data 2,383 volt dengan menggunakan batre dan 2,41 dengan menggunakan adaptor. Range ini masih termasuk logika high pada modul *Bluetooth* dengan level tegangan 0 – 3V.

Tegangan *Avcc* sama dengan tegangan input dari sistem yaitu , 4,04V dengan menggunakan batre dan 4,14V dengan menggunakan adaptor. Karena mikrokontroler ATMEGA8L ini bekerja hingga tegangan terendah sebesar 2,7V menurut data sheet. Namun attenuasi dari batre yang diijinkan adalah sebesar 0,5V saja. Dengan kata lain tegangan pada mode batre yang diijinkan adalah sampai dengan:

$$V_{batre} = 4,04 - 0,5 = 3,54\text{ V} \quad (4.28)$$

Selain untuk mencatu tegangan dari mikrokontroler pembatasan nilai atenuasi ini dilakukan untuk memperpanjang umur dari batre. Selain itu dapat dihitung *error* attenuasi batre adalah sebagai berikut.

- Mode batre

$$e_{Attenuasi} = \left| \frac{\left(\frac{2,006}{4,08}\right)-0,5}{0,5} \right| \times 100\% = 1,67\% \quad (4.29)$$

- Mode adaptor

$$e_{Attenuasi} = \left| \frac{\left(\frac{2,064}{4,14}\right)-0,5}{0,5} \right| \times 100\% = 0,29\% \quad (4.30)$$

Error yang kurang dari 2% pada hasil ukur terhadap spesifikasi yang dibuat dapat ditoleransi dan dapat dikatakan bahwa sistem yang dibuat sudah sangat mendekati dari spesifikasi yang telah ditentukan.

Beban total dari sistem dapat dihitung dengan cara menjumlahkan total beban dari sistem. mikrokontroler saat catu daya sebesar 4V adalah 6mA, *Bluetooth* HC05 40mA.

$$P_{uc} = 4,04 \times 6mA = 24,24mW \quad (4.31)$$

$$P_{bluetooth} = 4,04 \times 40mA = 161,6mW \quad (4.32)$$

$$I_{led} = 6 \times 1,6mA = 9,6mW \quad (4.33)$$

$$P_{led} = 4,04 \times 9,6mA = 38,78mW \quad (4.34)$$

$$R_{sensor} = 4,82mW \quad (4.35)$$

$$P_{total} = 24,24mW + 161,6mW + 9,6mW + 38,78mW + 4,82mW = 239,04mW \quad (4.36)$$

Tipe batre yang digunakan adalah batre li-ion dengan spesifikasi kapasitas daya sebesar 4,44Wh. Sehingga jika dikalkulasikan ketahanan dari batre saat mencatu sistem adalah sebagai berikut.

$$durability(jam) = \frac{4,44Wh}{239,04mW} = 18,57 \text{ jam} \quad (4.37)$$

Sehingga apabila batre dalam keadaan penuh maka sistem akan dapat dicatu selama 18,57jam.

-halaman ini sengaja dikosongkan

BAB V

PENUTUP

5.1 Kesimpulan

Setelah dilakukan pengujian pada sistem perangkat keras dan pengujian karakteristik dari sensor yang digunakan dapat diambil kesimpulan bahwa penerapan sistem *wireless* yang sudah diintegrasikan dengan robot EH1 Milano telah berhasil direalisasikan. Dengan tingkat validitas komunikasi modul *wireless* yang digunakan adalah 100% pada saat status *Bluetooth connected*. Jarak terjauh untuk melakukan komunikasi dengan menggunakan Bluetooth kurang dari 10 meter dengan halangan dan kurang dari 50 meter tanpa halangan dengan waktu pairing rata-rata 8 detik. Penggunaan *sensor flex* sebagai sensor untuk mendeteksi perubahan *bending* pada ruas jari dapat terukur lebih akurat dengan nilai *error* dari *trendline* (R^2) rata-rata sebesar 0,98618, *mean percentage error* terkecil 0,58% dan *error* tertinggi sebesar 15,13% yang menunjukkan bahwa sensor *flex* ini linier.

5.2 Saran

Beberapa saran yang dapat dilakukan untuk penelitian kedepan dari tugas akhir ini adalah menambahkan pendeteksian sinyal electromyography (EMG) untuk membandingkan inputan dari sensor *flex*. Selain itu kapasitas daya batre untuk mencatu sistem yang berada pada user hendaknya diperhatikan karena daya yang dikonsumsi sistem cukup besar, yaitu 239,04mW.

-halaman ini sengaja dikosongkan

LAMPIRAN

```
#include <mega8.h>
#include <stdlib.h>
#include <delay.h>
//#include <alcd.h>
#include <stdio.h>
#include <math.h>

#define ADC_VREF_TYPE 0x00

#define setFingerPos 68
#define getFingerPos 69
#define FastCalb 70
#define FirstCalb 66
#define Stop 65
#define Open 76

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input / (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA/=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA/=0x10;
    return ADCW;
}

#define pidmode !PINB.0
#define fc !PINB.1
#define monitoring !PINB.2

void sendData(char data)
{
    putchar(data);
}
```

```

}

void setfinger(char m_add,unsigned char position)
{
    sendData(setFingerPos);
    sendData(m_add);
    sendData(position);
}

data1[0]=read_adc(1);//baca data jempol
data[0]=data1[0]-310;

data1[1]=read_adc(5);//baca data tunjuk
data[1]=data1[1]-410;

data1[2]=read_adc(4);//baca data tengah
data[2]=data1[2]-390;

data1[3]=read_adc(3);//baca data manis
data[3]=data1[3]-355;

data1[4]=read_adc(2);//baca data kelingking
data[4]=data1[4]-375;

//mengset nilai nol samapi maksimal
if (data[0]<0) data[0] = 0;
else if (data[0]>135) data[0] = 135;

if (data[1] < 0) data[1] = 0;
else if (data[1]>115) data[1] = 115;

if (data[2] < 0) data[2] = 0;
else if (data[2]>115) data[2] = 115;

if (data[3] < 0) data[3] = 0;
else if (data[3]>110) data[3] = 110;

if (data[4] < 0) data[4] = 0;
else if (data[4]>120) data[4] = 120;

```

```
//normalisasi data sensor menjadi 0-255  
dat_sensor1[0]=((data[0]*1.888889)-255)*-1;  
dat_sensor2[0]=((data[1]*2.217391)-255)*-1;  
dat_sensor3[0]=((data[2]*2.217391)-255)*-1;  
dat_sensor4[0]=((data[3]*2.318182)-255)*-1;  
dat_sensor5[0]=((data[4]*2.125)-255)*-1;
```

-halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Prensilia, "*Extrinsic Robotic Hand Basic User Guide*", Prensilia s.r.l EH1 Milano Series, 2010.
- [2] Atmel, "*Atmel 8bit Avr Microcontroller ATMEGA8L datasheet*", Atmel Cooperation, 2013.
- [3] Spectrasymbol.com, "*Data Sheet Flex Sensor*", USA, 2014
- [4] Xin Ling, "*HC05 Data Sheet*", Guangzhou HC Information Technology Co., Ltd., 2011.
- [5] Lee Stan, "*Initial edition HC06*", 2013.
- [6] Nurmadyansyah, Rizky Fauzi, "*Perancangan dan Implementasi Sistem Kendali Robot Tangan Prensilia*", Jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember, 2013.
- [7] Pujiono, "*Rangkaian Elektronika Analog*", 2012.

-halaman ini sengaja dikosongkan

RIWAYAT PENULIS



Penulis memiliki nama lengkap Idil Fitrianto. Lahir di Bungo pada tanggal 24 April 1991. Penulis memulai pendidikan formal di SDN 108 Muara Bungo pada tahun 1997 - 2003. Melanjutkan pendidikan ke SMP Negeri 3 Muara Bungo hingga tahun 2006, dan melanjutkan pendidikan ke SMU N 2 Muara Bungo Hingga tahun 2009. Lulus SMA penulis melanjutkan jenjang pendidikannya di Universitas Gadjah Mada, Program Studi Teknik Elektro, bidang keahlian Teknik Elektronika. Setelah lulus Program Studi Diploma Teknik Elektro UGM pada tahun 2012, penulis melanjutkan pendidikan untuk meraih Strata-1 di jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember program Lintas Jalur, Bidang Studi Elektronika. Semasa kuliah penulis aktif ikut kegiatan robotika dari tahun 2010 - 2014, baik di UGM maupun di ITS.

Email :

idilfitrianto@gmail.com / idil12@mhs.ee.its.ac.id